Internet Applications

Dr./ Ahmed Mohamed Rabie

Chapter 2

Smart Objects Things in IOT

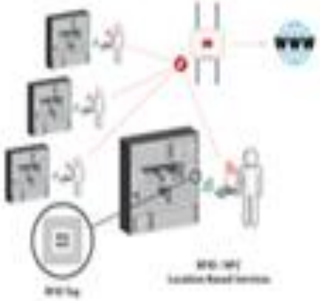
Smart objects are any physical objects that contain embedded technology to sense and/or interact with their environment in a meaningful way by being interconnected and enabling communication among themselves or an external agent.

Data Collection Devices

Smart Machinery

Phones and Tablets

Home Automation



RFID Systems

Digital Signage

Security Systems

Medical Devices

Sensors

A **sensor** does exactly as its name indicates: It senses. More specifically, a sensor measures some physical quantity and converts that measurement reading into a digital representation. That digital representation is typically passed to another device for transformation into useful data that can be consumed by intelligent devices or humans. Naturally, a parallel can be drawn with humans and the use of their five senses to learn about their surroundings.

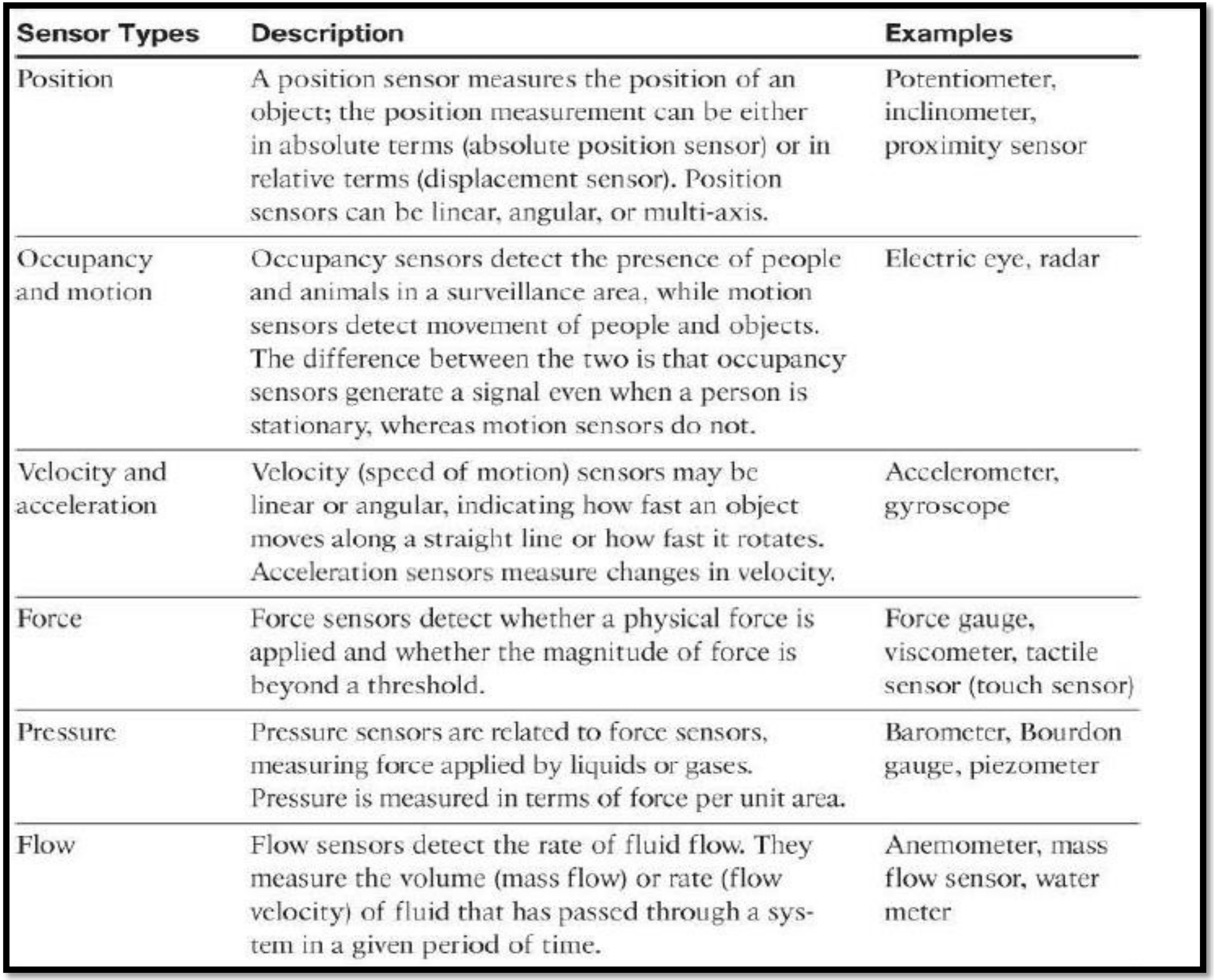
Human senses do not operate independently in silos. Instead, they complement each other and compute together, empowering the human brain to make intelligent decisions. The brain is the ultimate decision maker, and it often uses several sources of sensory input to validate an event and compensate for “incomplete” information.

There are a **number of ways to group and cluster sensors into different categories**, including the following:

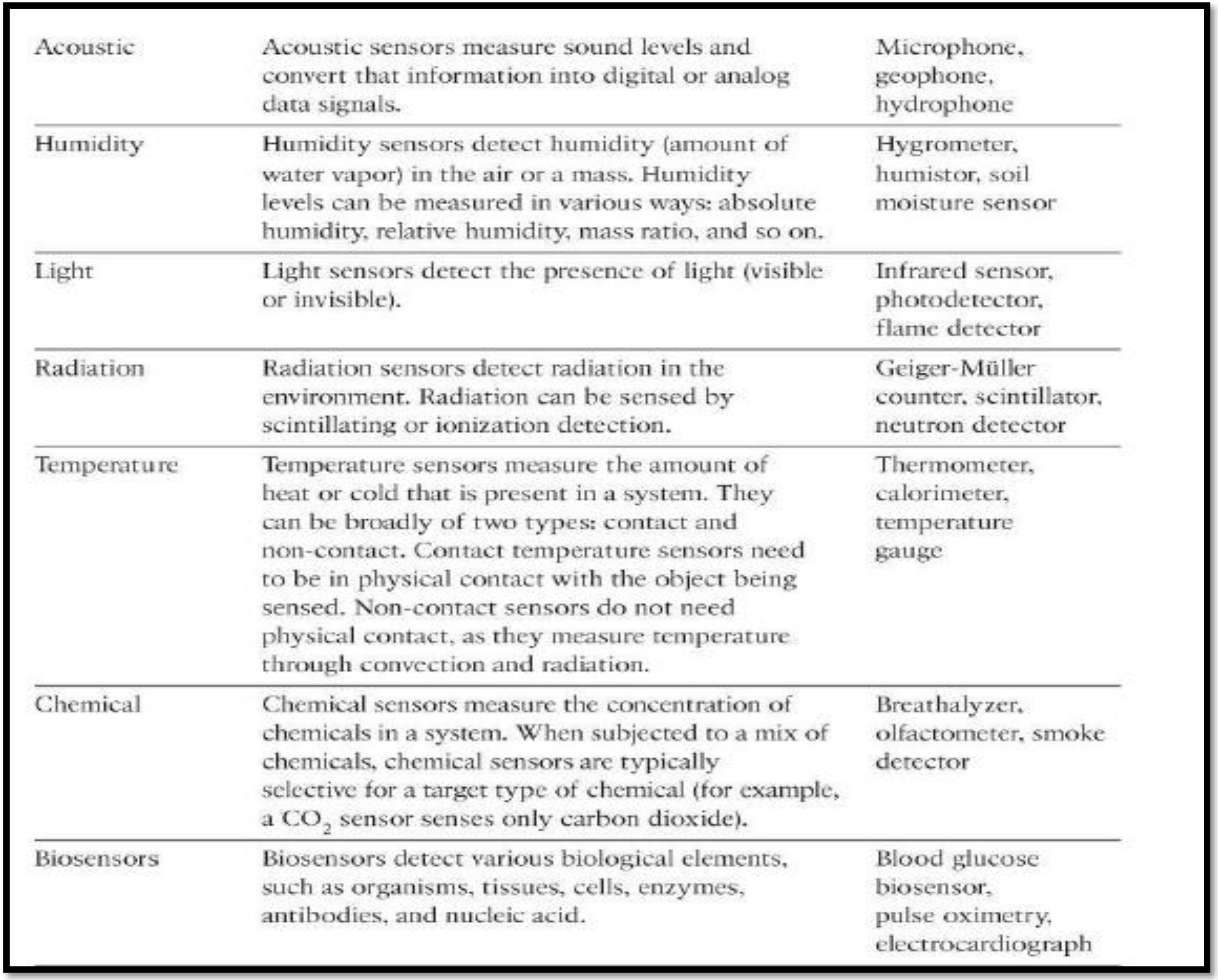
- Active or passive: Sensors can be categorized based on whether they produce an energy output and typically require an external power supply (active) or whether they simply receive energy and typically require no external power supply (passive).
- Invasive or non-invasive: Sensors can be categorized based on whether a sensor is part of the environment it is measuring (invasive) or external to it (non-invasive).

- Contact or no-contact: Sensors can be categorized based on whether they require physical contact with what they are measuring (contact) or not (no-contact).
- Absolute or relative: Sensors can be categorized based on whether they measure on an absolute scale (absolute) or based on a difference with a fixed or variable reference value (relative).
- Area of application: Sensors can be categorized based on the specific industry or vertical where they are being used.

- How sensors measure: Sensors can be categorized based on the physical mechanism used to measure sensory input (for example, thermoelectric, electrochemical, piezo resistive, optic, electric, fluid mechanic, photo elastic).
- What sensors measure: Sensors can be categorized based on their applications or what physical variables they measure.

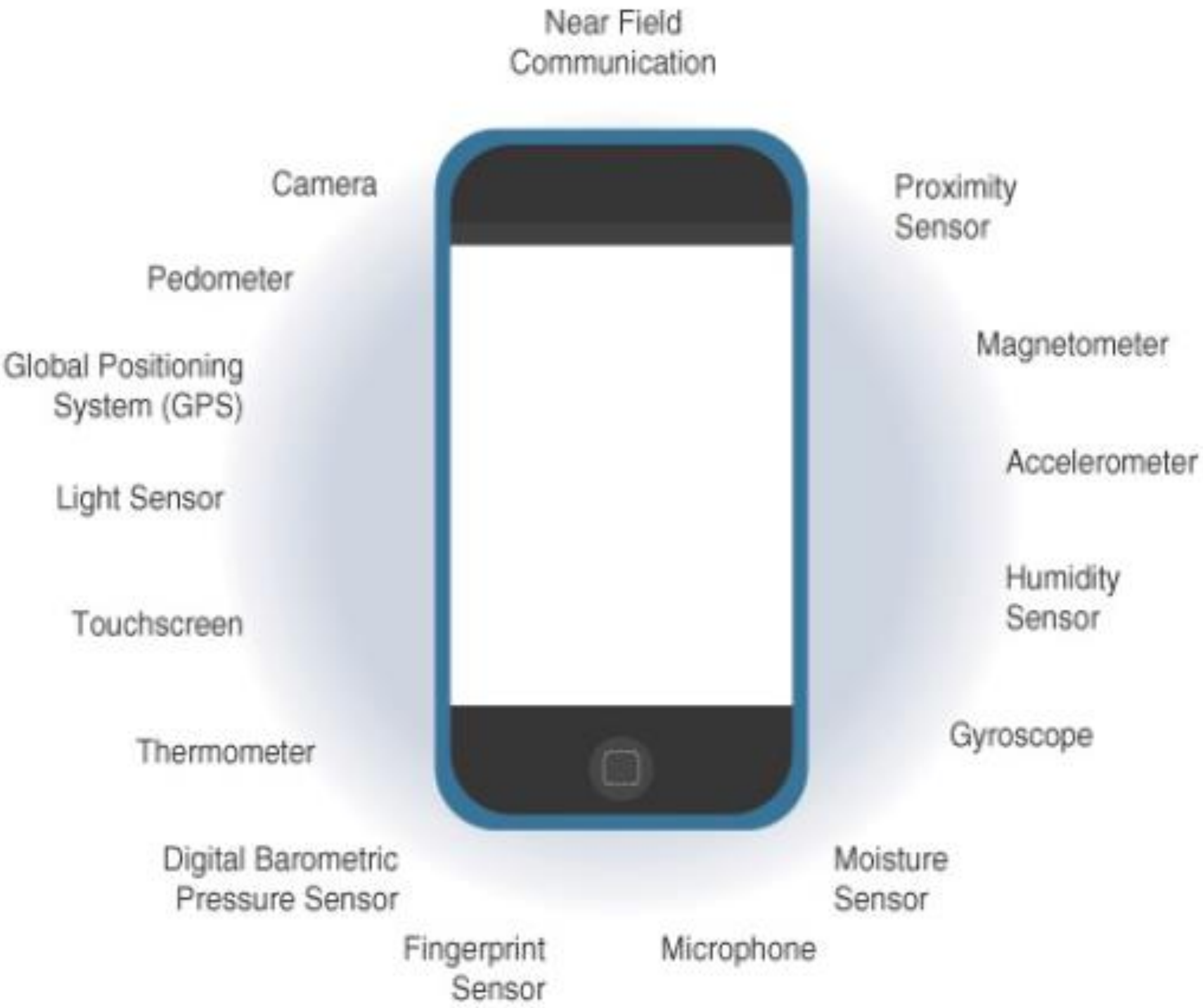


Sensor Types	Description	Examples
Position	A position sensor measures the position of an object; the position measurement can be either in absolute terms (absolute position sensor) or in relative terms (displacement sensor). Position sensors can be linear, angular, or multi-axis.	Potentiometer, inclinometer, proximity sensor
Occupancy and motion	Occupancy sensors detect the presence of people and animals in a surveillance area, while motion sensors detect movement of people and objects. The difference between the two is that occupancy sensors generate a signal even when a person is stationary, whereas motion sensors do not.	Electric eye, radar
Velocity and acceleration	Velocity (speed of motion) sensors may be linear or angular, indicating how fast an object moves along a straight line or how fast it rotates. Acceleration sensors measure changes in velocity.	Accelerometer, gyroscope
Force	Force sensors detect whether a physical force is applied and whether the magnitude of force is beyond a threshold.	Force gauge, viscometer, tactile sensor (touch sensor)
Pressure	Pressure sensors are related to force sensors, measuring force applied by liquids or gases. Pressure is measured in terms of force per unit area.	Barometer, Bourdon gauge, piezometer
Flow	Flow sensors detect the rate of fluid flow. They measure the volume (mass flow) or rate (flow velocity) of fluid that has passed through a system in a given period of time.	Anemometer, mass flow sensor, water meter



Acoustic	Acoustic sensors measure sound levels and convert that information into digital or analog data signals.	Microphone, geophone, hydrophone
Humidity	Humidity sensors detect humidity (amount of water vapor) in the air or a mass. Humidity levels can be measured in various ways: absolute humidity, relative humidity, mass ratio, and so on.	Hygrometer, humistor, soil moisture sensor
Light	Light sensors detect the presence of light (visible or invisible).	Infrared sensor, photodetector, flame detector
Radiation	Radiation sensors detect radiation in the environment. Radiation can be sensed by scintillating or ionization detection.	Geiger-Müller counter, scintillator, neutron detector
Temperature	Temperature sensors measure the amount of heat or cold that is present in a system. They can be broadly of two types: contact and non-contact. Contact temperature sensors need to be in physical contact with the object being sensed. Non-contact sensors do not need physical contact, as they measure temperature through convection and radiation.	Thermometer, calorimeter, temperature gauge
Chemical	Chemical sensors measure the concentration of chemicals in a system. When subjected to a mix of chemicals, chemical sensors are typically selective for a target type of chemical (for example, a CO_2 sensor senses only carbon dioxide).	Breathalyzer, olfactometer, smoke detector
Biosensors	Biosensors detect various biological elements, such as organisms, tissues, cells, enzymes, antibodies, and nucleic acid.	Blood glucose biosensor, pulse oximetry, electrocardiograph

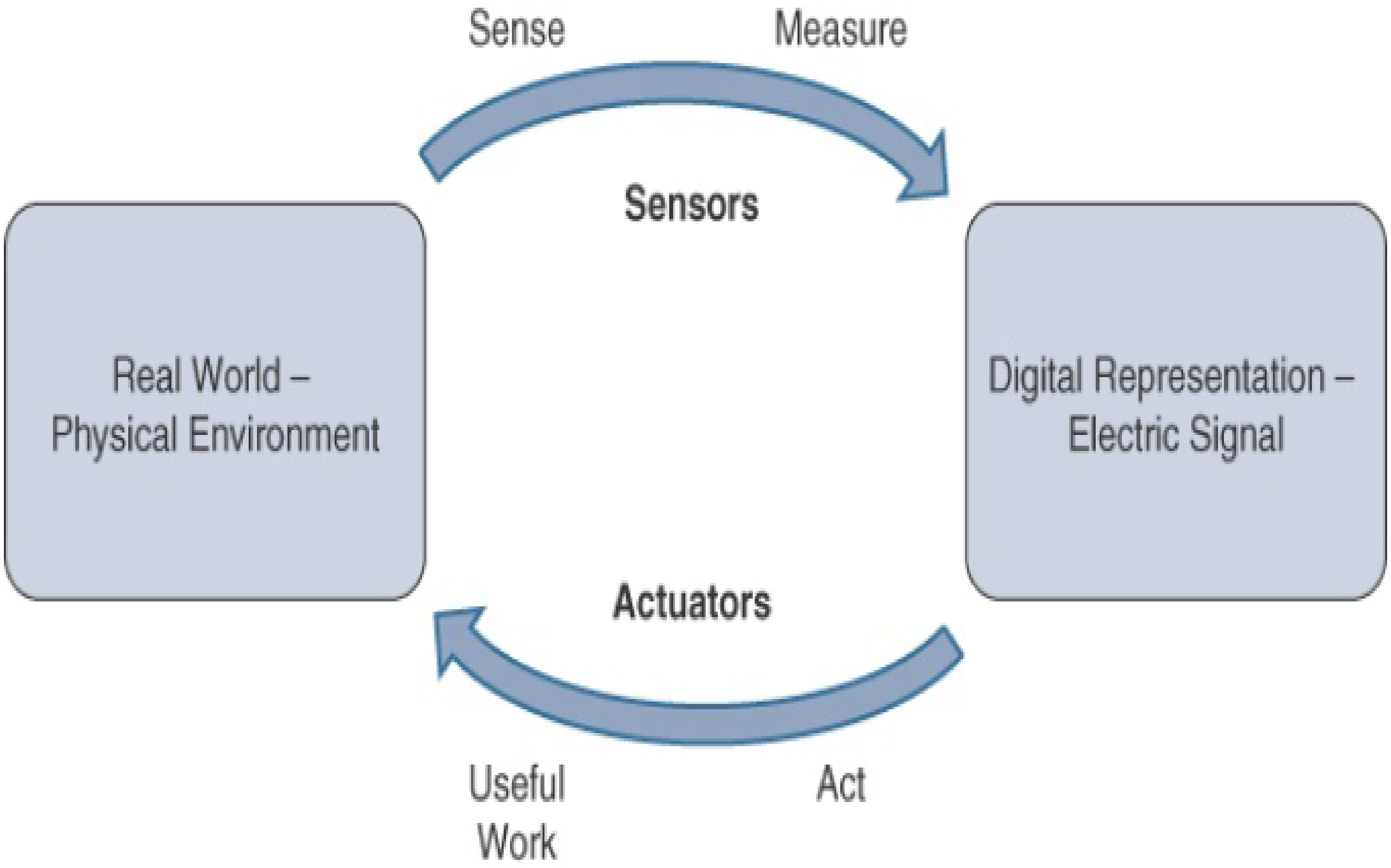
Perhaps the most significant accelerator for sensor deployments is mobile phones. More than a billion smart phones are sold each year, and each one has well over a dozen sensors inside it and that number continues to grow each year. Imagine the exponential effect of extending sensors to practically every technology, industry, and vertical. For example, there are smart homes with potentially hundreds of sensors، intelligent vehicles with +100 sensors each, connected cities with thousands upon thousands of connected sensors, and the list goes on and on.



Sensors in a Smart Phone

Actuators

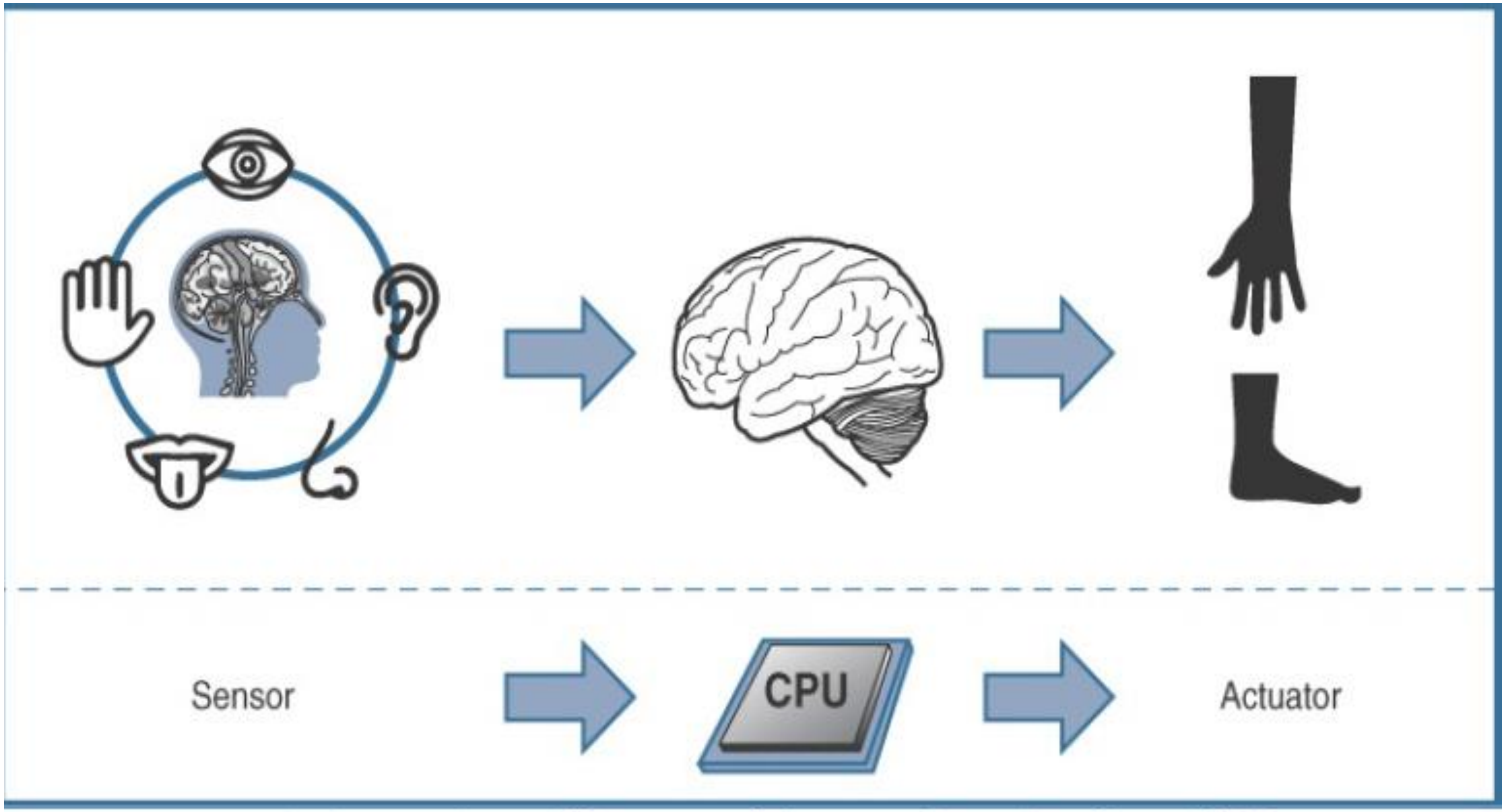
Actuators are natural complements to sensors. It demonstrates the symmetry and complementary nature of these two types of devices. Sensors are designed to sense and measure practically any measurable variable in the physical world. They convert their measurements (typically analog) into electric signals or digital representations that can be consumed by an intelligent agent (a device or a human). Actuators, on the others hand, receive some type of control signal (commonly an electric signal or digital command) that triggers a physical effect, usually some type of motion, force, and so on.



How Sensors and Actuators Interact with the Physical World

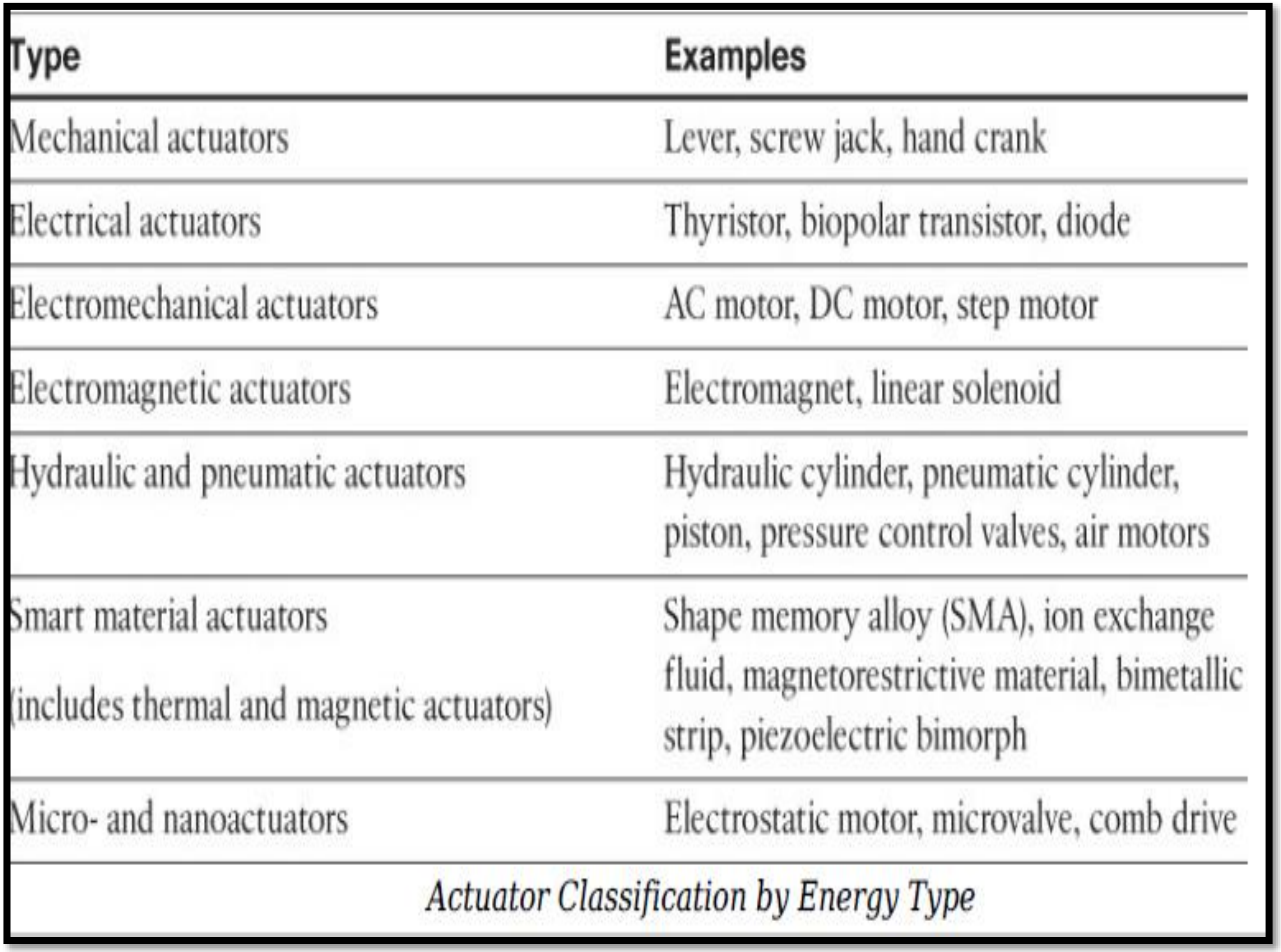
A parallel between sensors and the human senses. This parallel can be extended to include actuators. Humans use their five senses to sense and measure their environment. The sensory organs convert this sensory information into electrical impulses that the nervous system sends to the brain for processing. Likewise, IoT sensors are devices that sense and measure the physical world and (typically) signal their measurements as electric signals sent to some type of microprocessor or microcontroller for additional processing.

The human brain signals motor function and movement, and the nervous system carries that information to the appropriate part of the muscular system. Correspondingly, a processor can send an electric signal to an actuator that translates the signal into some type of movement (linear, rotational, and so on) or useful work that changes or has a measurable impact on the physical world. This interaction between sensors, actuators, and processors and the similar functionality in biological systems is the basis for various technical fields, including robotics and biometrics.



Comparison of Sensor and Actuator Functionality with Humans

Categorizing actuators is quite complex, given their variety, so this is by no means an exhaustive list of classification schemes. The most commonly used classification is based on energy type. Table shows actuators classified by energy type and some examples for each type. Again, this is not a complete list, but it does provide a reasonably comprehensive overview that highlights the diversity of function and design of actuators.

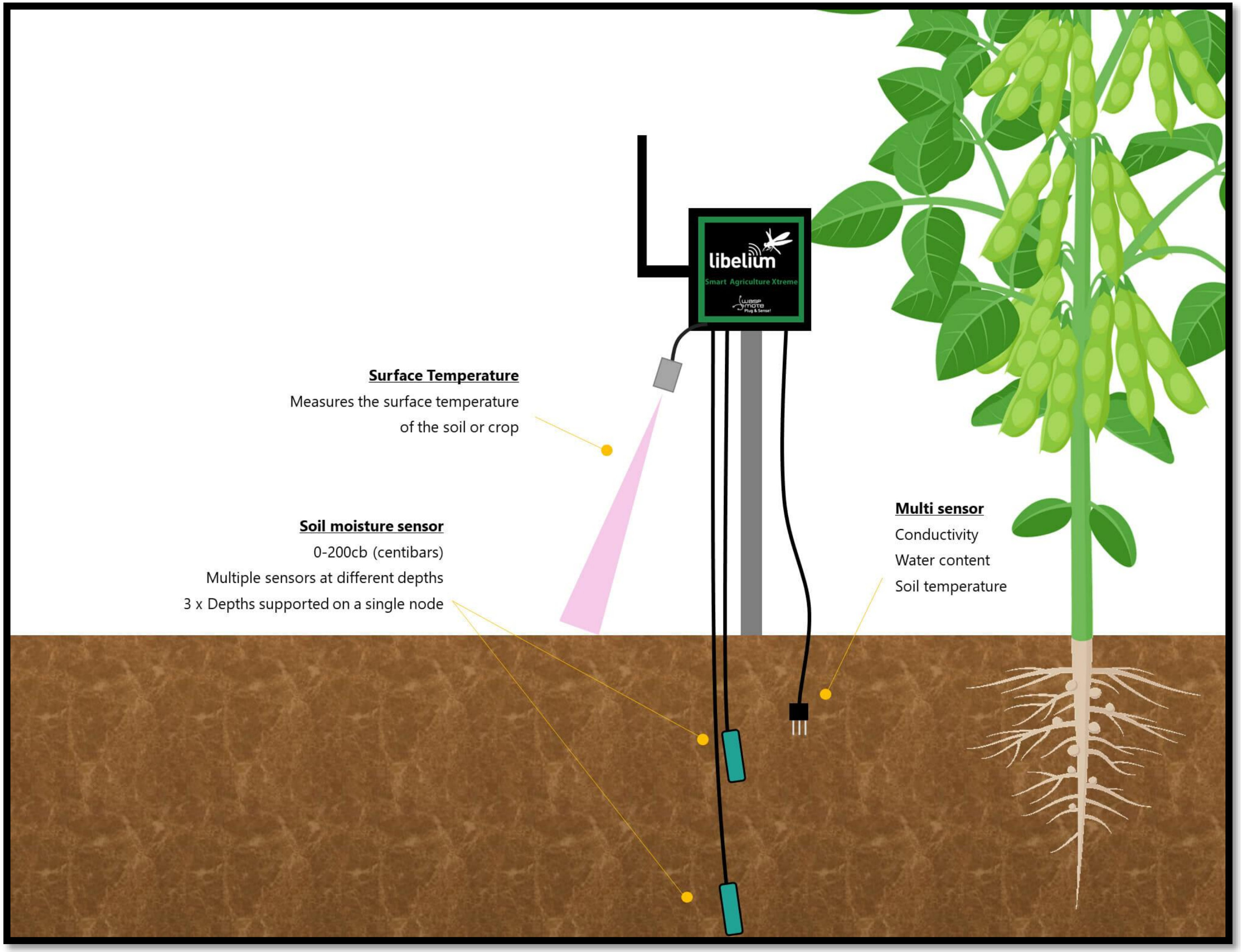


Type	Examples
Mechanical actuators	Lever, screw jack, hand crank
Electrical actuators	Thyristor, biopolar transistor, diode
Electromechanical actuators	AC motor, DC motor, step motor
Electromagnetic actuators	Electromagnet, linear solenoid
Hydraulic and pneumatic actuators	Hydraulic cylinder, pneumatic cylinder, piston, pressure control valves, air motors
Smart material actuators (includes thermal and magnetic actuators)	Shape memory alloy (SMA), ion exchange fluid, magnetorestrictive material, bimetallic strip, piezoelectric bimorph
Micro- and nanoactuators	Electrostatic motor, microvalve, comb drive

Actuator Classification by Energy Type

Whereas sensors provide the information, actuators provide the action. The most interesting use cases for IoT are those where sensors and actuators work together in an intelligent, strategic, and complementary fashion. This powerful combination can be used to solve everyday problems by simply elevating the data that sensors provide to actionable insight that can be acted on by work-producing actuators.

We can build on the precision agriculture example from the previous section to demonstrate how actuators can complement and enhance a sensor-only solution. For example, **the smart sensors used to evaluate soil quality** (by measuring a variety of soil, temperature, and plant characteristics) can be connected with electrically or pneumatically controlled valve actuators that control water, pesticides, fertilizers, herbicides, and so on. Intelligently triggering a high-precision actuator based on well-defined sensor readings of temperature, pH, soil/air humidity, nutrient levels, and so on to deliver a highly optimized and custom environment-specific solution is truly smart farming.



Surface Temperature

Measures the surface temperature
of the soil or crop

Soil moisture sensor

0-200cb (centibars)
Multiple sensors at different depths
3 x Depths supported on a single node

Multi sensor

Conductivity
Water content
Soil temperature

Smart Objects

Smart objects are, quite simply, the building blocks of IoT. They are what transform everyday objects into a network of intelligent objects that are able to learn from and interact with their environment in a meaningful way. It can’t be stressed enough that the real power of smart objects in IoT comes from being networked together rather than being isolated as standalone objects. This ability to communicate over a network has a multiplicative effect and allows for very sophisticated correlation and interaction between disparate smart objects.

Smart Objects: A Definition: Historically, the definition of a smart object has been a bit nebulous because of the different interpretations of the term by varying sources. To add to the overall confusion, the term smart object, despite some semantic differences, is often used interchangeably with terms such as smart sensor, smart device, IoT device, intelligent device, thing, smart thing, intelligent node, intelligent thing, ubiquitous thing, and intelligent product. In order to clarify some of this confusion, we provide here the definition of smart object.

A **smart object**, is a device that has, at a minimum, the following **four defining characteristics:-**

- **Processing unit**: A smart object has some type of processing unit for acquiring data, processing and analyzing sensing information received by the sensor(s), coordinating control signals to any actuators, and controlling a variety of functions on the smart object, including the communication and power systems. The specific type of processing unit that is used can vary greatly, depending on the specific processing needs of different applications. The most common is a **microcontroller** because of its small form factor, flexibility, programming simplicity, ubiquity, low power consumption, and low cost.

- **Sensor(s) and/or actuator(s):** A smart object is capable of interacting with the physical world through sensors and actuators. A sensor learns and measures its environment, whereas an actuator is able to produce some change in the physical world. A smart object does not need to contain both sensors and actuators. In fact, a smart object can contain one or multiple sensors and/or actuators, depending upon the application.

- **Communication device:** The communication unit is responsible for connecting a smart object with other smart objects and the outside world (via the network). Communication devices for smart objects can be either **wired or wireless**. Overwhelmingly, in IoT networks smart objects are wirelessly interconnected for a number of reasons, including cost, limited infrastructure availability, and ease of deployment. There are myriad different communication protocols for smart objects.

- **Power source:** Smart objects have components that need to be powered. Interestingly, the most significant power consumption usually comes from the communication unit of a smart object. As with the other three smart object building blocks, the power requirements also vary greatly from application to application. Typically, smart objects are limited in power, are deployed for a very long time, and are not easily accessible.

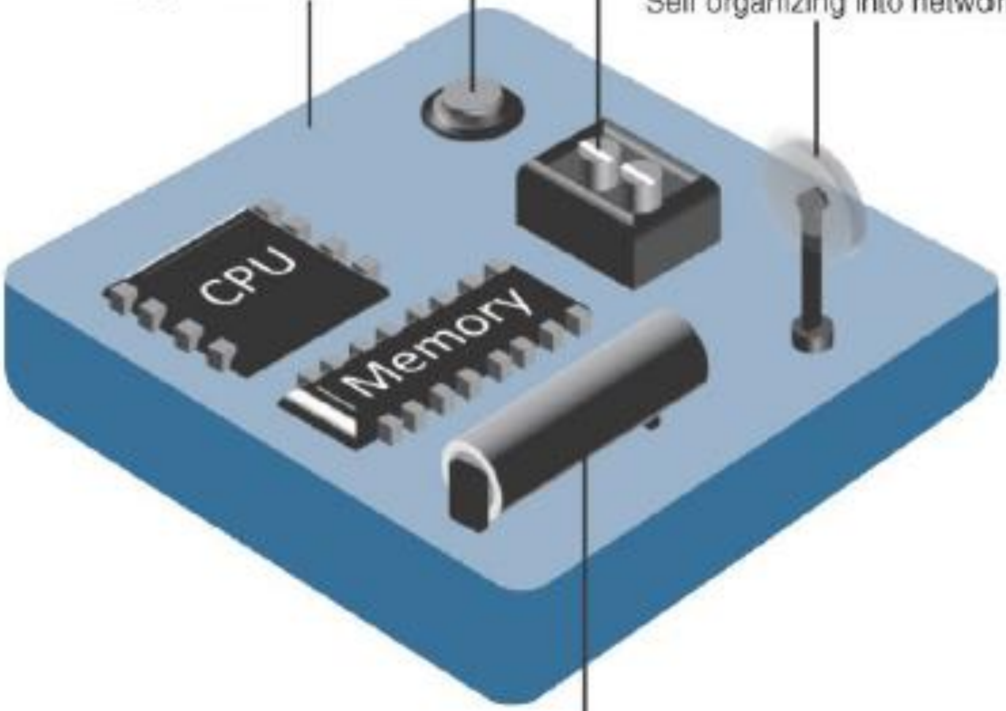
This combination, especially when the smart object relies on battery power, implies that power efficiency, judicious power management, sleep modes, ultra-low power consumption hardware, and so on are critical design elements. For long-term deployments where smart objects are, for all practical purposes, inaccessible, power is commonly obtained from scavenger sources (solar, piezoelectric, and so on) or is obtained in a hybridized manner, also tapping into infrastructure power.

Sensor
That can measure physical data
(temperature, vibration, pollution...)

Actuator
Capable of performing a task
(change traffic lights, rotate a mirror...)

Tiny Low Cost Computer
Embedded into objects to make them smart
Can be organized into networks

Communication Device
Receives instructions, sends or routes data
Self organizing into networks



Power Source
Scavenger (solar/wind), battery, mains

Characteristics of a Smart Object

Trends in Smart Objects

As this definition reveals, it is perhaps variability that is the key characteristic of smart objects. They vary wildly in function, technical requirements, form factor, deployment conditions, and so on. Nevertheless, there are certain important macro trends that we can infer from recent and planned future smart object deployments. Of course, these do not apply to all smart objects because there will always be application dependent variability.

These are broad smart objects generalizations and trends impacting IoT:

- **Size is decreasing**: There is a clear trend of ever-decreasing size. Some smart objects are so small they are not even visible to the naked eye. This reduced size makes smart objects easier to embed in everyday objects.
- **Power consumption is decreasing**: The different hardware components of a smart object continually consume less power. This is especially true for sensors, many of which are completely passive. Some battery-powered sensors last 10 or more years without battery replacement.

- **Processing power is increasing:** Processors are continually getting more powerful and smaller. This is a key advancement for smart objects, as they become increasingly complex and connected.
- **Communication capabilities are improving**: It's no big surprise that wireless speeds are continually increasing, but they are also increasing in range. IoT is driving the development of more and more specialized communication protocols covering a greater diversity of use cases and environments.

- **Communication is being increasingly standardized**: There is a strong push in the industry to develop open standards for IoT communication protocols. In addition, there are more and more open source efforts to advance IoT.

These trends in smart objects begin to paint a picture of increasingly sophisticated devices that are able to perform increasingly complex tasks with greater efficiency. A key enabler of this paradigm is improved communication between interconnected smart objects within a system and between that system and external entities (for example, edge compute, cloud). The power of IoT is truly unlocked when smart objects are networked together in sensor/actuator networks.

Sensor Networks

Sensor Networks: A sensor/actuator network **(SANET),** as the name suggests, is a network of sensors that sense and measure their environment and/or actuators that act on their environment. The sensors and/or actuators in a SANET are capable of communicating and cooperating in a productive manner. Effective and well-coordinated communication and cooperation is a prominent challenge, primarily because the sensors and actuators in SANETs are diverse, heterogeneous, and resource-constrained.

SANETs offer highly coordinated sensing and actuation capabilities. Smart homes are a type of SANET that display this coordination between distributed sensors and actuators. For example, smart homes can have temperature sensors that are strategically networked with heating, ventilation, and air-conditioning (HVAC) actuators. When a sensor detects a specified temperature, this can trigger an actuator to take action and heat or cool the home as needed.

While such networks can theoretically be connected in a wired or wireless fashion, the fact that SANETs are typically found in the “real world” means that they need an extreme level of deployment flexibility. For example, smart home temperature sensors need to be expertly located in strategic locations throughout the home, including at HVAC entry and exit points.

The following are some advantages and disadvantages that a wireless-based solution offers:

Advantages:

- Greater deployment flexibility (especially in extreme environments or hard-to-reach places)
- Simpler scaling to a large number of nodes
- Lower implementation costs
- Easier long-term maintenance
- Effortless introduction of new sensor/actuator nodes
- Better equipped to handle dynamic/rapid topology changes

Disadvantages:

- Potentially less secure (for example, hijacked access points)
- Typically lower transmission speeds
- Greater level of impact/influence by environment
- Not only does wireless allow much greater flexibility, but it is also an increasingly inexpensive and reliable
- technology across a very wide spectrum of conditions even extremely harsh ones.

Wireless Sensor Networks

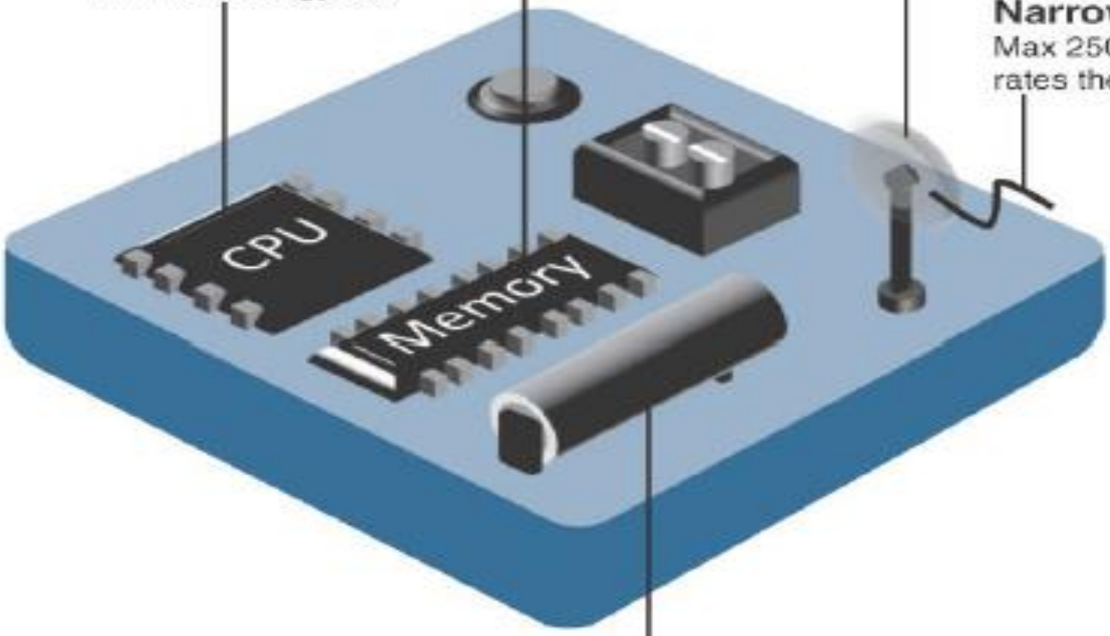
Wireless sensor networks are made up of wirelessly connected smart objects, which are sometimes referred to as motes. The fact that there is no infrastructure to consider with WSNs is surely a powerful advantage for flexible deployments, but there are a variety of design constraints to consider with these wirelessly connected smart objects. Figure illustrates some of these assumptions and constraints usually involved in WSNs.

Low Memory
Few tens of kilobytes
Embedded OS
(TinyOS, Contiki etc...)

Lossy Communications
Low Power Wireless mesh predominantly IEEE802.15.4
Also IEEE P1902.1 (Power Line Comms)

Moderate CPU Power
Minimize energy use

Narrowband Media
Max 250KB/s, lower
rates the norm



Power Consumption Is Critical
Energy efficiency is paramount
Battery powered devices must last years

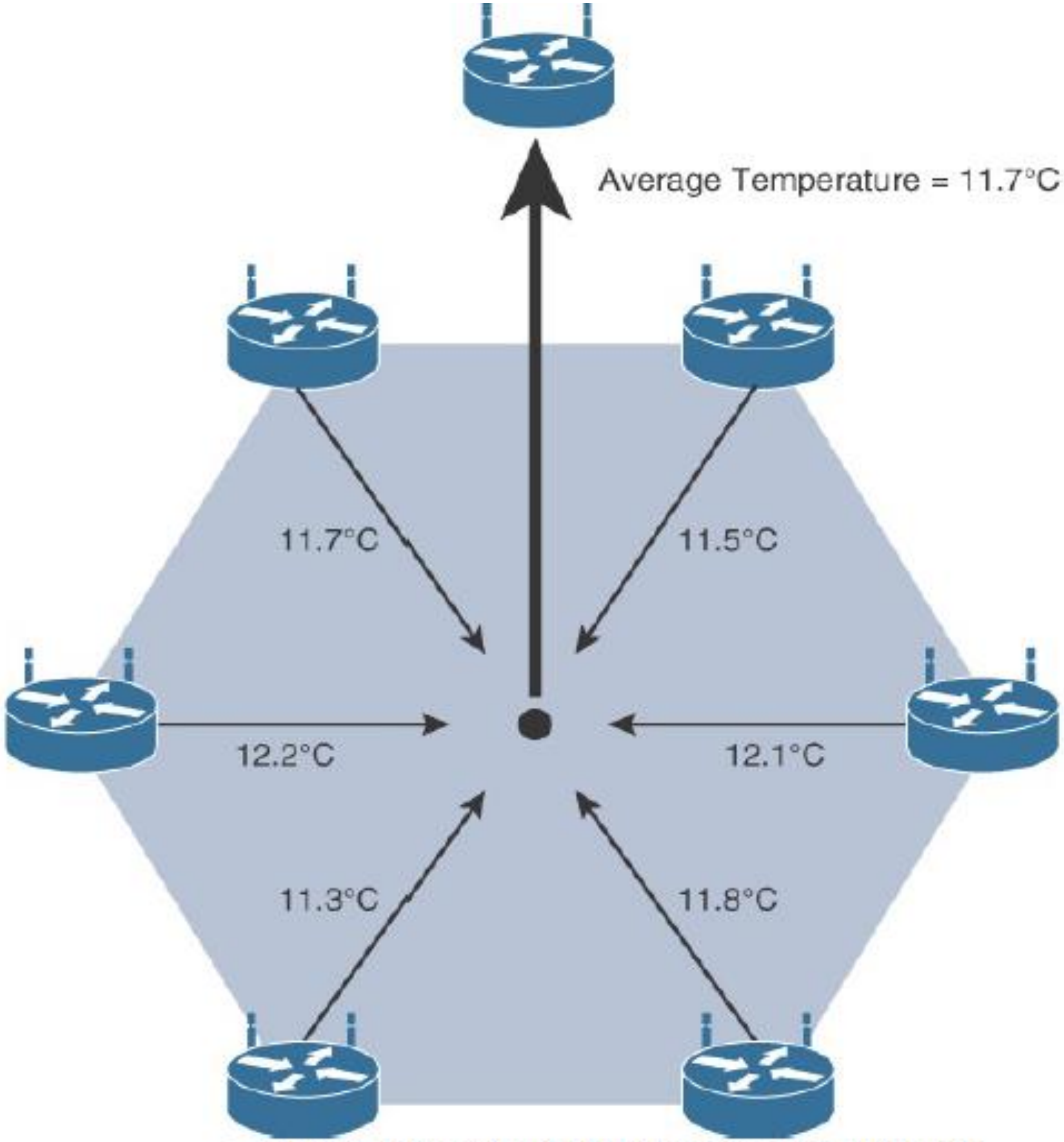
Design Constraints for Wireless Smart Objects

The following are some of the most significant limitations of the smart objects in WSNs:

- Limited processing power
- Limited memory
- Lossy communication
- Limited transmission speeds
- Limited power

These limitations greatly influence how WSNs are designed, deployed, and utilized. The fact that individual sensor nodes are typically so limited is a reason that they are often deployed in very large numbers. As the cost of sensor nodes continues to decline, the ability to deploy highly redundant sensors becomes increasingly feasible. Because many sensors are very inexpensive and correspondingly inaccurate, the ability to deploy smart objects redundantly allows for increased accuracy.

Such large numbers of sensors permit the introduction of hierarchies of smart objects. Such a hierarchy provides, among other organizational advantages, the ability to aggregate similar sensor readings from sensor nodes that are in close proximity to each other. Figure shows an example of such a data aggregation function in a WSN where temperature readings from a logical grouping of temperature sensors are aggregated as an average temperature reading.



Data Aggregation in Wireless Sensor Networks

These data aggregation techniques are helpful in reducing the amount of overall traffic (and energy) in WSNs with very large numbers of deployed smart objects. This data aggregation at the network edges is where fog and mist computing, "IoT Network Architecture and Design," are critical IoT architectural elements needed to deliver the scale and performance required by so many IoT use cases. While there are certain instances in which sensors continuously stream their measurement data, this is typically not the case.

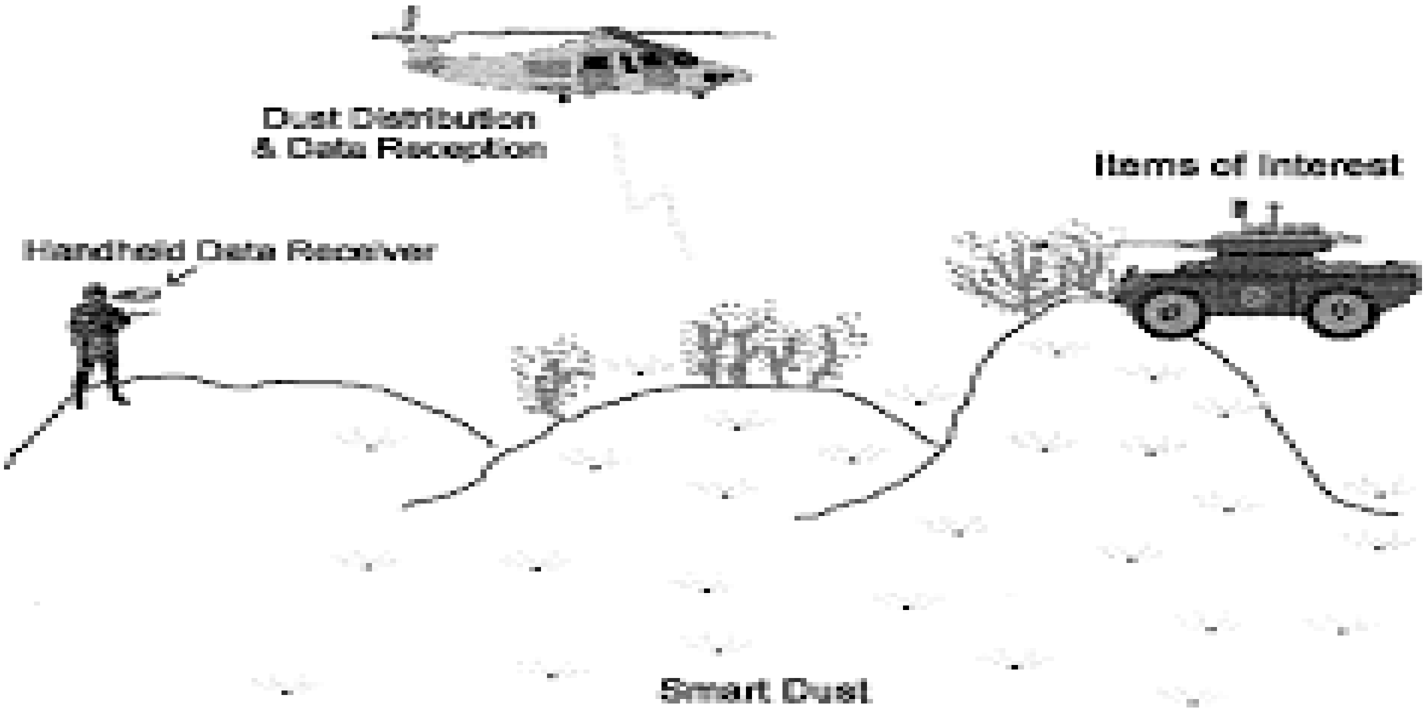
Wirelessly connected smart objects generally have one of the following **two communication patterns**:

- **Event-driven:** Transmission of sensory information is triggered only when a smart object detects a particular event or predetermined threshold.
- **Periodic:** Transmission of sensory information occurs only at periodic intervals.

The decision of which of these communication schemes is used depends greatly on the specific application. For example, in some medical use cases, sensors periodically send postoperative vitals, such as temperature or blood pressure readings. In other medical use cases, the same blood pressure or temperature readings are triggered to be sent only when certain critically low or high readings are measured.

As WSNs grow to very large numbers of smart objects, there is a trend toward ever-increasing levels of autonomy. For example, manual configuration of potentially thousands of smart objects is impractical and unwieldy, so smart objects in a WSN are typically self-configuring or automated by an IoT management platform in the background. Likewise, additional levels of autonomous functions are required to establish cohesive communication among the multitudinous nodes of large-scale WSNs that are often ad hoc deployments with no regard for uniform node distribution and/or density.

For example, there is an increasing trend toward “**smart dust**” applications, in which very small sensor nodes are scattered over a geographic area to detect vibrations, temperature, humidity, and so on. This technology has practically limitless capabilities, such as military (for example, detecting enemy troop movement), environmental (for example, detecting earthquakes or forest fires), and industrial (for example, detecting manufacturing anomalies, asset tracking). Some level of self-organization is required for networking the scads of wireless smart objects such that these nodes autonomously come together to form a true network with a common purpose.



Smart dust particles

smartdust on a finger

This capability to self-organize is able to adapt and evolve the logical topology of a WSN to optimize communication (among nodes as well as to centralized wireless controllers), simplify the introduction of new smart objects, and improve reliability and access to services. Additional advantages of being able to deploy large numbers of wireless low-cost smart objects are the inherent ability to provide fault tolerance, reliability, and the capability to extend the life of a WSN, especially in scenarios where the smart objects have limited battery life.

Autonomous techniques, such as self-healing, self-protection, and self-optimization, are often employed to perform these functions on behalf of an overall WSN system. IoT applications are often mission critical, and in large-scale WSNs, the overall system can’t fail if the environment suddenly changes, wireless communication is temporarily lost, or a limited number of nodes run out of battery power or function improperly.

Communication Protocols for WSNs

There are literally thousands of different types of sensors and actuators. To further complicate matters, WSNs are becoming increasingly heterogeneous, with more sophisticated interactions. This heterogeneity is manifested in a variety of ways. For instance, WSNs are seeing transitions from homogenous wireless networks made up of mostly a single type of sensor to networks made up of multiple types of sensors that can even be a hybridized mix of many cheap sensors with a few expensive ones used for very specific high-precision functions.

WSNs are also evolving from single-purpose networks to more flexible multipurpose networks that can use specific sensor types for multiple different applications at any given time. Imagine a WSN that has multiple types of sensors, and one of those types is a temperature sensor that can be flexibly used concurrently for environmental applications, weather applications, and smart farming applications.

Coordinated communication with sophisticated interactions by constrained devices within such a heterogeneous environment is quite a challenge. The protocols governing the communication for WSNs must deal with the inherent defining characteristics of WSNs and the constrained devices within them. For instance, any communication protocol must be able to scale to a large number of nodes. Likewise, when selecting a communication protocol, you must carefully take into account the requirements of the specific application and consider any trade-offs the communication protocol offers between power consumption, maximum transmission speed, range, tolerance for packet loss, topology optimization, security, and so on.

Wireless sensor networks interact with their environment. Sensors often produce large amounts of sensing and measurement data that needs to be processed. This data can be processed locally by the nodes of a WSN or across zero or more hierarchical levels in IoT networks. Communication protocols need to facilitate routing and message handling for this data flow between sensor nodes as well as from sensor nodes to optional gateways, edge compute, or centralized cloud compute. IoT communication protocols for WSNs thus straddle the entire protocol stack. Ultimately, they are used to provide a platform for a variety of IoT smart services.

As with any other networking application, in order to interoperate in multivendor environments, these communication protocols must be standardized. This is a critical dependency for IoT and one of the most significant success factors. IoT is one of those rare technologies that impacts all verticals and industries, which means standardization of communication protocols is a complicated task, requiring protocol definition across multiple layers of the stack, as well as a great deal of coordination across multiple standards development organizations.

Recently there have been focused efforts to standardize communication protocols for IoT, but, as with the adoption of any significant technology movement, there has been some market fragmentation. While there isn’t a single protocol solution, there is beginning to be some clear market convergence around several key communication protocols.