Virtual Computing

The Virtual

For something to be virtual, it must have some type of correspondence with something physical. A virtual dog, for example, needs to have a correspondence of some sort with a dog or dogs. That correspondence may exist in the mind of the person, as when you see a picture of a dog or see the word "dog" in a book, a magazine or on a computer screen. The dog that you see in your mind is the result of the context in which you see the picture or read the word. It is easy to think that the dog in the book is not physical, but it is. The virtual nature of the dog in your mind is directly linked to the physical nature of your brain and nervous system. They in turn connect the dog in your mind to the dog as represented by the ink on the paper, or the Braille imprints under your fingertips, or the sound waves entering your ears, or the photons entering your eye from the page or computer monitor. The picture of the dog or the word "dog" serves as the focal point or hub for a great many connections to other things, some of which are physical in their relationship to the picture or the word and some of which are virtual. In all cases, the virtual exists side-by-side with the physical.

A person might say, "But the word 'dog' is itself a virtual representation of a physical dog, a signifier and not a signified." That is true to a point. If there is a physical dog somewhere that someone simulates in a computer game, we can say that the computer game is a simulation or virtual representation of that particular dog. But we can also say that the virtuality of the dog in our mind is the result of the physicality of the dog in the computer game. It is like an equation of the form (V)irtuality approximates (P)hysicality or $V \approx P$. Virtuality is always the result of the virtual on one side and the physical on the other. Depending on where you stand in any given chain of virtual and physical relationships, one person's virtual can be another person's physical, and vice versa. Even within the confines of your own preference for one taxonomy over another, you may find there are several different ways to look at the virtual nature of an object.

For example, you may perceive the relationship of the virtual to the physical as linear, meaning a virtual dog approximates a physical dog, or even that a series of virtual dogs must approximate and include a series of physical dogs of the form $V \approx (P / V) \approx (P / V)$, etc., where the leftmost side is always a V and the rightmost side is always a P. That is one way to look at it. You may also perceive the virtual as hierarchical in nature, meaning the physicality of the dog is a single thing that begins to differentiate as we break it up into pieces. The whole dog becomes its head, body, legs and feet, hair, ears, eyes, nose, bark, friendliness (or not) and so forth. Just like Linnaeus, we create a tree of categorizations that we call "dog" and it is this tree that we refer to when describing our virtual dog, either in our minds or in our computer simulation. You may perceive the relationship of the physical to the virtual in many other ways as well: as a set of relationships that depend on the ideas of groups, members and the relationships between those groups and members. This greatly increases the number of moving parts, because to understand or simulate or virtualize a dog, you may need to include groups and members of groups that are not dogs or parts of dogs. Things such as bones, chewed shoes, loyalty, chasing cats and cars, and peeing on fire hydrants, bushes and couch cushions come to mind.

It is important to understand that all philosophical, logical and mathematical concepts of reality begin with the ideas of naming and partitioning. When you name a thing the very act of naming it, separates it from everything else. Once you have named a number of things you separate them into groups. By separating them into groups you partition them. From that point on, you have to decide which things, as they are named, belong to which groups. The physical and the virtual are two names describing two groups. The physical refers to matter and energy. The virtual refers to information. But we know that information, language, always has a material aspect. And we also know that the physical always has an informational aspect.

The question is this. Does the virtual precede and determine (i.e, control) the physical, or does the physical precede and determine the virtual? It is not an easy question to answer.

Most people are comfortable with the idea that the physical precedes, determines and controls the virtual. But what if we turn it around and allow the virtual to control the physical? What does that mean? It means that virtuality, as much as physicality, relies on the idea of cause and effect.

Cause and Effect

To say that one thing causes another is to say several things. The first thing it says is that there is more than one thing in the universe, an obvious conclusion to some, not so to others. The second thing it says is that at least one of these things precedes the other or others in time. The third thing it says is that at least one of these things shares a relationship with the others that requires that changes in the first thing will effect changes in one or more of the other things. And finally, it says that to be separated in time is to be separated in space. That any change in an object is both a change in time and a change in space.

This leads us to an interesting question. When an object changes, how do other objects affected by that change find out about it? They have to be told somehow. How else is the first object's "cause" going to result in any "effect" on the parts of the other objects? One way they can find out is by being directly connected to the first thing, like cogs and gears in a watch. Another way they can find out is to be told by an intermediary or messenger of some sort, say a subatomic particle traveling at the speed of light from one object to another. A third way they can find out is to just instantaneously know about it somehow, without being told anything at all, even if the "cause" object is at the opposite end of the universe from the "effect" object. No message. No messenger. Nothing.

These are the three types of cause and effect. One, a local model universe in which changes in one thing are the result of changes in only those other things that are directly connected to it. Two, a message model universe in which things are perceived as either objects or events. The object is static. The event is change. The event can then be perceived as a message between two objects. Three is the quantum universe in which things can be in two or more places at once, but only if we do not look at them directly. In this quantum world, everything exists as information before it exists as matter and energy. In this world, objects are not bound by conventional rules of space and time. A thing can exist in a coherent (i.e, knowable) superposition (many places at once). A change at any superposition will result in an instantaneous change at all the others.

Virtual Computing

It is not a question of which of these models of cause and effect are true. They are each true in their own way. They represent the history and evolution of the virtual. They also represent the path that computing has taken in its pursuit of the virtual. Traditional computers rely on things being directly connected to each other or connected to each other via messages. Quantum computers take advantage of the ability to know the state of many things in the quantum world by knowing the state of only a few things in the everyday physical world.

That is one of the ultimate goals of computing: to create virtual images of the physical universe at every level of its physicality and then to use those images to control it.

Over time, as more and more physical things have their corresponding virtual images represented in computers, it becomes easier to effect changes in the physical things by effecting changes in their corresponding virtual images. Since virtual images of the physical world can operate and interact with each other at something near the speed of light, the corresponding physical objects they represent can also be made to interact with each other at greater levels of detail and at greater speeds and greater physical distances.

Each and every computer ever created has the same five basic components: I/O (Input and Output) connections, a message bus, a processor, memory and a program that it executes. This is as true for a cell phone as it is for a supercomputer. This concept of a computer, first described by Alan Turing, has not changed in over sixty years. What has changed is the speed, size, heterogeneity and location of the computer and its parts. Whereas the computer was once thought of primarily as a physical device, it is now increasingly thought of as a logical device in which every physical component has its counterpart in software. With the emergence of a global information network, it is now possible to think of the computer, not only in terms of its physical nature, but as a collection of software components each of which is virtual and each of which corresponds in some way to one of the computer's five basic components.

Distributed Virtual Computing

The following discussion is taken from the book, Red Moon:

As more and more hardware functions are implemented in software and hardware becomes more ubiquitous, the computer is becoming an abstract entity whose characteristics are more organic than mechanical. This new paradigm may be described as virtual computing – an environment in which all hardware functions have been emulated in software and traditional software systems (including operating system, communication, presentation, application and data logic) become virtual in nature, maintaining an existence independent of any particular hardware configuration.

The key concept underlying this emerging paradigm is that of a virtual system. Virtual systems are normally conceived of as computing environments which take input in the form of physical movements via a "glove" or some other device and use that information to drive a system which provides visual output in the form of "virtual reality." That is one way to think about virtual systems – as an interactive system in which the physical environment drives, controls or otherwise enables the visualization of a virtual environment. But there is another way of defining virtual systems. It is possible for the virtual environment to control the physical environment. This way of thinking about virtual systems leads to the following definition:

A virtual system is any system or model that:

1. is primarily conceptual in nature, having its origin in language and language-based artifacts,
2. is a fully functional image or representation of a physical system and
3. controls the organization and operation of one or more such physical systems.

A physical system may in turn underlie one or more virtual systems.

Virtual systems are not a new idea in computing. They have existed for years in the form of virtual machines. A virtual machine usually takes the form of a virtual processor that is a software image of a physical processor. A virtual processor normally has access to memory, stores instructions and data in its registers and processes instructions in much the same way as a physical processor. It is common for a single physical processor or CPU to support multiple virtual processors.

Virtual machines do not have to take the form of virtual processors. They can be operating system kernels, database engines or knowledge domain specific applications. In fact, any type of software can be implemented as a virtual machine. The key advantage of virtual machines is that they can be made hardware independent. Programs written for a virtual machine do not have to be rewritten if they are moved to a different hardware or software platform as long as the virtual machine has been migrated to that environment.

Once a computing environment is abstracted in this way the next step is the creation of distributed virtual machines. A distributed virtual machine may be defined as a virtual machine (such as a virtual processor) that can be replicated and distributed across heterogeneous computing environments while maintaining a single logical image.

To better understand the concept of a distributed virtual machine, it is helpful to review the history of the virtual machine itself. The virtual machine was first described by Alan Turing and its first physical manifestation was the mainframe. To this day, with the exception of the quantum computer first described by Richard Feynman (the universe itself is a quantum computer), all computing models including rulebase systems and artificial neural networks can be reduced to Turing machines.

The mainframe began as a big box located at a single point in space and was used by the military during World War II to solve mathematical problems. In time, the box was replicated and used by various organizations and companies to solve many types of problems. The principal of the mainframe has never changed. Put all of your eggs (i.e. your data and decision making processes) in one basket and guard that basket with your life.

Over time computing power increased dramatically. During the same period the cost and size of the computer decreased just as dramatically. The general purpose computer eventually took the form of the PC. This led to client-server computing, a form of computing that allowed the data to be stored on a central computer, while the problem solving occurred on the desktop.

Then came the Internet. The Internet allows any computer to communicate with any other computer anywhere in the world. Computers connected in this way can share both data and processes. But what is still missing is the ability to move the data and processes from any place to any other place at any time in real time and to treat many different computers, processes and data sets as a logical unity. The virtual system that is assembled and/or created should be able to maintain a close correspondence with and coordinate the interactions of many physical systems. This collection of computers, programs and data should be thought of in the same way that a conductor thinks of the musicians that comprise the orchestra during the performance of a symphony.

Distributed virtual machines by definition contain the intelligence to replicate and to place themselves in those (hardware and software) environments that are most conducive to the accomplishment of their goals and purposes. Distributed virtual machines in turn provide a means of creating virtual computing environments.

Virtual computing is defined as a computing paradigm that:

1. implements virtual systems in software,
2. operates independently of hardware operating system, network and programming language constraints and
3. allows traditional transactional models (e.g. client, server, agent) to be implemented as distributed virtual machines.

Simply stated, virtual computing consists of one or more virtual systems managed by one or more distributed virtual machines within a network computing environment.

In the evolution of computing, traditional computing architectures will eventually be replaced by systems that do not impose arbitrary hardware and software hierarchies based on traditional transactional models. This new computing paradigm may seem too radical a departure from traditional computing models to be easily implemented within existing computing environments. The fact is, virtual computing can and does co-exist with existing systems and the conversion to virtual computing can occur incrementally and in parallel with existing systems.

Virtual computing enables virtual systems. In the near future distributed virtual machines and distributed virtual systems will become the dominant metaphor by which all social artifacts, whether they are political, economic or cultural, will be defined.