Yes And No

By Kevin A. Sensenig
Draft 1.01
2018 December 3

Yes and no are mutually interdependent, because of the object and object-space they're talking of.

They're just references, pointers to actual object, object-space, or reality as one takes in action.

They are points in logical space, within the context of logical statements and logical space. They may have representational value, but they at the same time are just other points in logical space, and in the context of logical statements. This is supported by Wittgenstein in "Tractatus Logico Philosophicus", and reflects his philosophy; and it is supported even his introductory statement "The world is all that is the case."

To take yes/no as being the reality itself is a mistake. One can get caught in false dualities. One can become stuck in thought, or attached to the yes/no, or to either one, unable to see the domain object and object-space. One can get caught in discrimination, as seeing the objects and things and people before one as strictly individual, separable, and distinct: without the neither being-nor-non-being that they are (the nondualistic view).

But to talk of the domain-space itself is so important.

This represents the Zen view.

Even in computer programming with an if {} else {} statement, the branch is just 1) an acknowledgment of some truth state or object state; and 2) a pointer or directive to another truth statement or object method (object and object space), or a do- or reflect-something. You can see this in Objective-C, or Java, or even in C.

Even the digital transistor is really binary arithmetic, not a yes/no. It is in 2 states only because that's the extent of binary arithmetic, but if we had a stable, state decimal transistor (decimal arithmetic) we'd probably use it instead: floating point would be decimal and without approximation error to its place, and it would be a deeper density of computing.