

Conclusions

I. Preliminaries

A. Review
1. Last time we talked about **higher-order digitality**: discreteness at the of concepts or types
2. We said—though this would take much more time to explore in any depth—that the *non-higher-order-discreteness* of many computational concepts has enormous implications for the scientific study of computation-in-the-wild.

B. Independence
1. One thing we can already see is that separating out discreteness from other issues in computing is beneficial.
2. That is, breaking digitality out into its own construal frees up the others to deal with their respective topics (materiality, semantic participation, etc.) without distraction.
3. It should also be clear that the issues of digitality cross cut the other construals.
4. This is true even first-order digitality.
   a. For example, wrt FSM, we talked briefly, last week, about how one can understand both symbols and their referents to be digital or continuous. And that these were essentially independent (one could have discrete or continuous symbols, and—independently—discrete or continuous referents).
   b. Similarly, we pointed out that the current (official) "theory of computation" was a mathematical theory of causality—again, in a digital or discrete version. (The work on continuous Turing machines, etc., can be viewed as an attempt to generalise to a continuous version.)
5. The same orthogonality holds, wrt higher-order digitality. When we now go back and summarize what we have learned about computing, we will be able to ask, about each construal, whether its issue or subject matter was expressed in (higher-order) discrete form— and if so, whether that ultimately proved a help or a hindrance.

C. Plan
1. With these remarks, I want to conclude our third critique, of the DSM construal of computation (which takes computers to be digital state machines).
2. It is time to step back and pull together what we have seen, over the course of the semester, in a series of conclusions. This will wrap up our investigation.
3. We will do this in two stages:
   a. Today, we will talk about things we can conclude about *computing per se.*
   b. Next week, we will talk about the *consequences* of these conclusions. This will involve us in asking what we can conclude about the nature of the *world more generally*, given what we have learned, here, about computing, in particular.

II. Conclusions · I — Overall

A. Computing is an admixture of material or **mechanism** and **meaning** or semantics.
B. The "mind/body" problem for machines—we took that as the fundamental dialectic.
C. That has proved durable
   1. I *still* think computing is a mixture of the material and semantic.
   2. Nothing we have seen, in the course of our three critiques, has raised any consideration suggesting we abandon that basic stance towards the subject matter.

III. Conclusions · II — Specific

A. Per construal
   1. FSM
      a. Yes, computing is constituted by an interplay of the two main aspects (the FSM construal was the only one to recognise this explicitly).
      b. *Participation:* whereas the formal view took them to be separate, computers are physically (effectively) and semantically (non-effectively) *involved* with their subject matters.
      c. Various boundaries—between inside and out, symbols and their referents, continuity and discreteness, the (relatively) concrete and the (relatively) abstract, etc.—*cross-cut*, and are *gradual*
      d. The combination of this gradualist participatory picture has enormous consequences.
         i. For example, in cognitive science, it meshes with recent depictions of the mind as embodied and (intimately) embedded in the world.
         ii. It also means that many staples of our traditional theoretic imaginations—such as the notion of a "brain in a vat," and the solipsist's question of whether there is a world "out there"—are based on fundamental miscomprehensions.[1]
   2. EC
      a. *Effectiveness:* an (inchoate) mathematical theory of the flow of effect
      b. Essentially a mathematical theory of *causality*
      c. As such, promises an answer to half of the original puzzle: the *body* or *materiality* side (or at least it may do that, once massively revised—maybe merged with physics, expanded to deal with metric time, side-effects, and on-going behavior, etc.)
   3. DSM
      a. *First-order:* what matters about digital computing—the digital revolution, etc.—is not that the systems we build are discrete (many are not), but that they are digitally *implemented*
         i. This allows them to avoid the ravages of decay and environmental buffeting
      b. First-order digitality is achieved by imposing (and holding a system to) an *abstraction* over its underlying physical states, which not only (i) ignores the inevitable micro state-internal variation, but also, and more miraculously, (ii) confines that internal variation to stay *within the confines* of the digital abstraction.

---

[1] The point is not just that nothing in the *computational* theory of mind suggests that such conceptualisations are true. The moral (though we haven't had time in this course to explore it) is stronger: the considerations we raised against these "inside-outside" views of computing would hold, equally, for the mind.

c. *Higher-order:* many important high-level computational notions (object-oriented, distributed, virtual machine, security, etc.) are *not higher order discrete.*

d. This has methodological ramifications with respect to the potential scientific nature of the field

B. Some other morals

1. **Implementation:** many critical computational properties do not cross implementation boundaries
2. **Theoretical status:** Many (most?) notions in computer science have not yet been adequately theorised. Just a few (there are many others):
    - a. Process
    - b. Programming
    - c. Implementation
    - d. Abstraction
    - e. *Process* semantics (as opposed to programming *language* semantics)
3. **Meaning & mechanism:** Fundamental dialectic remains in play, in terms of which three main results can be identified:
    - a. Semantics
        - i. Remains outstanding, without a theory
        - ii. It is (in general) *non-effective* (cf. conclusion of first critique)
    - b. Materiality: best hope is the reconstructed version of EC, above
    - c. The two strands—effective materiality and non-effective semantics—are *intertwined*

C. There are more, but this gives a feel …

IV. Conclusions · III — Overall

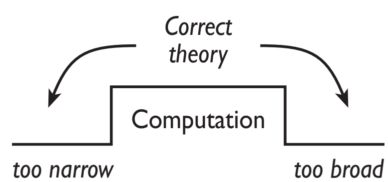

Figure 1 — Two ways to fail

A. Status

1. Should we leave it at that?
2. No! To do so—as I suggested last time—would be to miss the most important result of the investigation
3. That result has to do with higher-order digitality (introduced last time), and its relation to the very notions and distinctions we have been working with all semester

B. Higher-order discreteness (and its lack)

1. Go back to the basic question: about a *comprehensive theory of computing*
2. At the beginning, said there were two ways a candidate theory could fail (see figure 1):
    - a. Too *narrow* (false of extant computers)
    - b. Too *broad* (true of every material thing)
3. Look at each construal, focusing on how, in its original formulation, it *steered clear of the "too broad" cliff* (i.e., what it said, about computing, that suggested that computing was "less than all there is"):
    - a. **FSM:** syntax (effectiveness) *independent of semantics*
    - b. **EC:** a theory of computability *independent of material substrate*
    - c. **DSM:** a theory of computing in which *everything is digital (clear and sharp)*

4. In each case, what distinguishes the construal is a claim that some distinction is *clear and precise* (absolute)
   a. **FSM:** between the *syntax* (effective workings) and the *semantics*
   b. **EC:** between the *calculation of functions* and the underlying *material substrate*
   c. **DSM:** between *everything* and *everything else*
5. That is:

> **C1** Each construal took *computing to be* **special** in virtue of exemplifying a form of *higher-order discreteness* with respect to *its particular issue.*

6. But then look at how our criticism went. In each case, we started out in rough agreement. That is:

> **C2** In each case, we accepted, as fundamental (to computing), the issue that the construal was ultimately determined to be dealing with.

7. Examples
   a. For example, wrt the FSM construal, we took
      i. *Topic:* interplay between effective and semantic
      ii. *Insight:* the two realms, instead of being sharply (discretely) distinguished, were implicated in each other. Also: that systems didn't work, by and large, by the causal efficacy of semantic properties (since semantic properties are at least sometimes, and perhaps always, impotent).
   b. Similarly for the EC construal:
      i. *Topic:* flow of effectiveness—what it takes, and how hard it is, for one piece or region of the physical world to be rearranged into some other arrangement, in a systematic way depending on what happens to it
      ii. *Insight:* there are limits to what can be done, and regularities about how hard it is for things that can be done, that are (relatively) independent of the level of abstraction at which the machine is analysed.
8. Diagnosis
   a. But then what was the trouble?
   b. The trouble was that the constitutive distinction, in terms of which the construal was formulated, was taken (by that construal) to be *too sharp and absolute* to do justice to reality (that is: computation in the wild).
   c. E.g., for FSM, we said "Sure enough, syntax and semantics are *partly separate*. That is a real insight into the nature of reality. But they aren't *completely independent*.
   d. Similarly, for EC, we said "Yes, that's right, there are things to be said (mathematically characterised) about the potential behavior of material devices, at very high levels of abstraction. But not *completely* independent of facts about material implementation.
9. That is:

> **C3** In each case, the ultimate trouble was that computation—computation in the wild, especially—was *not higher-order discrete*, after all.

10. But now think of what that means. The very thing that made computers **special**—the fact that computers (allegedly) deal with issues that every semantical (interpretational) system in the world deals with, but in their own special way—is exactly what we ended up finding failed to be true. That is:

> **C4** Systematically, our inquiries failed to sustain the idea that computers are special after all!

11. I.e., there is a sense in which the *phenomenon itself* is falling off a kind of "too broad" cliff:
    a. Every attempt to restrict a theory, so that *it* isn't too broad (in the sense of applying to any materially significant entity), ends up being *too narrow to capture the phenomenon.*
    b. What seems to be true is that the *phenomenon can't be contained*.

C. Formality
1. This can be stated, much more compactly, in terms of formality
2. Each construal, at a very general level, can be characterized as follows:
    a. It deals with a distinctive topic
    b. It is based on an important insight, framed in terms of some notion or distinction
    c. It treats that distinction as higher-order discrete
    d. In each case, that higher-order discreteness is called **formality**
3. That is, we can understand formality as a form of (or name for) higher-order discreteness

> **C5** Formality = (a generalised form of) higher-order discreteness

4. Given this analysis, we can summarize the results of our investigations as follows:

> **C6 Computation-in-the-wild is not formal**

5. In a single sentence, that distills the results of the whole investigation (of these construals, and other others that we didn't have time to look at).
6. But that is not all. Because the point the of the first moral (**C1**) above was that computation was *assumed to be formal.* In fact, more specifically:

> **C7** What has classically been taken to distinguish computation (computers) from general intentional (semantic) systems, has been their (alleged) formality.[2]

---

[2]Cf. Fodor's famous "formality condition"—an expression of this general thesis, in the context of the first (FSM) construal.

7. Together, **C6** and **C7** generate another version of **C4**:

C8 Computers are not special.

8. On Thursday, we will explore the (non–trivial) consequences of this last statement.

—— *end of file* ——