

Ontología del *software*: contra la teoría de la realizabilidad múltiple

Matías Nicolás Cristini*



102-120

Resumen

En este trabajo realizamos un análisis de la condición ontológica de los objetos digitales. Para hacerlo, desarrollamos una exposición argumental de cuatro momentos. En primer lugar, presentamos la teoría de la realizabilidad múltiple del *software* de Blanco y Berti (2016). En segundo lugar, proponemos un escenario problemático para el cual la teoría en cuestión se muestra insuficiente como insumo explicativo. En tercer lugar, consideramos dos contraargumentos posibles: el primero está fundamentado en el modelo procedimental de la comunicación de

Abstract

In this work, we carry out an analysis of the ontological condition of digital objects. To do so, we develop a four-moment argumentative exposition. First, we present Blanco and Berti's (2016) theory about the multiple realizability of software. Second, we propose a problematic scenario for which the theory in question appears to be insufficient as an explanatory input. Third, we consider two possible counter-arguments: the first one relies on Bootz's (2011) procedural model of communication and the second one, on the problematic link between physi-

* Universidad Nacional de Mar del Plata. Correo electrónico: matiascristini@gmail.com.

Bootz (2011) y el segundo, en el vínculo problemático entre entidades físicas y modelos. Por último, mostramos cómo dichos contraargumentos resultan infundados y proponemos una teoría de la identidad físico-digital, así como un reemplazo del concepto de *medio digital asociado* por el de *equipo tecnológico* para explicar el carácter relacional de la ontología de los objetos digitales.

Palabras clave

realizabilidad múltiple
software
ontología

cal entities and models. Finally, we show how these counter-arguments are ill founded and we propose a theory of physical-digital identity, as well as a replacement of the concept of *associated digital medium* in favor of that of *technological equipment* to explain the relational character of the ontology of digital objects.

Keywords

multiple realizability
software
ontology

Fecha de recepción

13 de diciembre de 2021

Aceptado para su publicación

28 de mayo de 2022

Introducción: dos perspectivas contemporáneas respecto a la ontología del *software*

En este trabajo nos proponemos tratar el estado ontológico de los objetos digitales, es decir, del *software*¹. Para ello, nos resultará útil un abordaje inicial de dos formas en las que se lo caracteriza actualmente: las interpretaciones *folk* y académica. A riesgo de dar un diagnóstico demasiado general, podríamos decir que ambas abrevan en una base neo-cartesiana. El “neo” refiere a la aplicación del viejo dualismo de sustancias, que versaba sobre mente y cuerpo, a la oposición contemporánea *software* vs. *hardware*². Varios rasgos de la dualidad original subsisten en su versión actualizada. Por ejemplo, en esta última suele representarse a lo virtual (otro nombre para lo digital-*software*) como escindido de su soporte físico o dirigiéndolo.

La primera variante de esta perspectiva neo-cartesiana forma parte del discurso del usuario común de PC, celulares y otros dispositivos computacionales (de allí su nombre: *folk*). Este discurso alude al *software* mediante metáforas de corte etéreo (como “la nube” para referir a internet) o descripciones mentalistas (como “se quedó pensando” para referir al proceso de carga en una PC). Se refiere a los sitios web, por ejemplo, como si subsistiesen en algún plano inmaterial al cual meramente se accede mediante un aparato físico.

Javier Blanco y Agustín Berti mencionan una interpretación de la ontología del *software* de corte académico que podríamos pensar como una respuesta a esta primera perspectiva (Blanco y Berti, 2016: 198-199). Si bien la postura académica comparte, a grandes rasgos, los presupuestos de la postura *folk* en torno a la inmaterialidad del *software*, concluye que, justamente por ellos, hace falta negar su existencia efectiva. Es decir, esta segunda perspectiva propone extirpar los objetos digitales del campo de lo que es por considerar que implican necesariamente una reintroducción (indeseada) del dualismo cartesiano. Entre las alternativas de interpretación que se proponen desde la perspectiva académica los autores nombran el fisicalismo, el semanticismo, el mecanicismo y otras varias (Blanco y Berti, 2016: 198).

Una tercera perspectiva: la realizabilidad múltiple

Frente a la posición *folk* y a las múltiples variantes académicas que rechazan el *software* de plano, Blanco y Berti (2016) proponen entenderlo como una entidad

¹ Nos referiremos a este tipo de entidad con ambos términos de forma indistinta.

² Para un análisis histórico-conceptual del dualismo en cuanto teoría de la mente, cfr. Robinson (2020).

que siempre requiere soporte físico, pero que es múltiplemente realizable. Esta posición permitiría “materializar” el *software* y, al mismo tiempo, resolver los problemas de identidad que surgen al pensar su existencia en relación con el *hardware*. Concretamente, los autores se apoyan en dos conceptos de Matthew Kirschenbaum (2012): materialidad formal y materialidad forense³. Aclaran que la materialidad forense

se basa en el principio de individualización que permite cada inscripción de un código en un dispositivo de almacenamiento dado, a partir de la idea de que no puede haber dos cosas exactamente idénticas en el mundo físico (en principio desde el momento en que no pueden ocupar el mismo espacio al mismo tiempo) (Blanco y Berti, 2016: 202).

Por otro lado, la materialidad formal consistiría en “la ocurrencia de una serie discreta de variables dentro de un rango dado, que permitiese identificar dos ocurrencias de una serie delimitada de signos como idénticas” (Blanco y Berti, 2016: 202). Basándonos en conceptos utilizados comúnmente en relación con la teoría⁴ de la realizabilidad múltiple, podríamos decir que estas dos formas de materialidad se relacionan con el par tipo-caso. Un tipo sería una clase general de entidad, mientras que un caso sería una instancia particular y concreta de un tipo⁵. Por ejemplo, la serie de letras *A, A* contiene dos casos (instancias) del tipo *A* (la letra entendida generalmente). Podríamos decir, entonces, que, en los términos de Kirschenbaum (y de Blanco y Berti), las instancias físicas particulares de materialidad forense contarían como casos varios de realización de un mismo tipo (reconocible como único a partir de la ocurrencia de una serie discreta de variables dentro de un rango dado).

La presentación inaugural de la realizabilidad múltiple se le debe a Hilary Putnam (1960). A continuación, consignamos una reformulación sumaria de su argumento, según sus premisas y conclusión (referido a la múltiple realización de tipos mentales en tipos físicos):

- P1: Al menos algunos tipos mentales (por ejemplo, dolor) son realizables por distintos tipos físicos (activación de fibras-C, etc.).
- P2: Si un tipo mental es realizable por distintos tipos físicos, entonces no

³ Si bien recurren a estos conceptos, más adelante veremos que les achacan ciertas limitaciones.

⁴ Cuando hablamos de la teoría de la realizabilidad múltiple usamos el término “teoría” en un sentido lato para referirnos a la posición filosófica que tiene dicho nombre.

⁵ Para una discusión más pormenorizada respecto al par tipo-caso, cfr. Wetzel (2018).

- puede ser idéntico a ninguno de ellos.
- C: Al menos algunos tipos mentales no son idénticos a ningún tipo físico específico (Bickle, 2020).

Nótese que la relación signada en este razonamiento es tipo-tipo, no tipo-caso (se habla de tipos mentales y tipos físicos), pero eso no supone un problema en la medida en que cada tipo mental y tipo físico se realizan, efectivamente, en casos (cada instancia particular de dolor se realiza en una instancia particular de activación de fibras-C u otro tipo de proceso físico). Lo importante es la relación entre un tipo de proceso y otro: dos tipos de procesos físicos distintos pueden realizar un mismo tipo de proceso mental.

En este punto, cabe hacer una aclaración: si bien Blanco y Berti (2016), basándose en Kirschenbaum (2012), plantean una relación de solo dos niveles de materialidad (formal y forense), es necesario pensar una relación tipo-tipo entre materialidades formales no explicitada para abordar su propuesta desde la formulación canónica de la realizabilidad múltiple. Esto se debe a que proponen como evidencia de la realizabilidad múltiple del *software* el hecho de que un mismo objeto digital (supongamos, el archivo foto.jpg) sea copiable en soportes con materialidades forenses radicalmente distintas, como *pendrives*, *floppy disks*, etc. Pero, a menos que se proponga que cada uno de esos casos físicos pertenece, a su vez, a tipos físicos distintos, la formulación de la realizabilidad múltiple previamente enunciada no aplicaría. Hace falta postular la existencia de los tipos *pendrive*, *floppy disk*, etc. para luego arribar a la conclusión de que el tipo foto.jpg es realizable en todos ellos. Cada *floppy disk* particular que contenga el archivo foto.jpg será, a su vez, un caso del tipo *floppy disk*, pero, en principio, es necesario distinguir entre los distintos tipos físicos que permiten la realización múltiple del objeto digital, así como es necesario rastrear la identidad de este objeto digital a la materialidad formal que, a su vez, es realizable a partir de todos ellos. Especificando más, podríamos decir que el tipo de materialidad formal mediante el cual puede realizarse un archivo foto.jpg en un *floppy disk* se da en cuanto una relación o disposición determinada de polaridades magnéticas, mientras que en un CD se da mediante otra de inscripciones láser. A su vez, habrá varios casos de las materialidades formales del soporte (varios *floppy disks* o CD) en los que podrán realizarse varios casos de la materialidad formal del código (varias copias de foto.jpg). Ocurre algo similar con el planteo de Putnam: lo que la teoría debe demostrar no es que el dolor, por ejemplo, puede realizarse en varios casos de activación de fibras-C (puesto que esto sería coherente con una teoría que identifique el dolor con dicha activación física), sino que puede realizarse en varios *tipos* de eventos físicos (activación de fibras-C y otros tipos de actividad neurofisiológica).

Esto último es lo que explica la posibilidad de entender varias instancias del archivo foto.jpg como copias, aun si están inscriptas en medios físicos radical-

mente distintos: el tipo *foto.jpg* se identifica con la serie de variables discretas dentro de un rango dado que puede realizarse en varios tipos de soportes materiales concretos (mediante polaridades magnéticas, inscripciones láser, etc.). Si no aceptamos esto y, por el contrario, identificamos como materialidad formal únicamente la serie de variables que hacen a cada soporte físico particular, nos veríamos en la necesidad de afirmar que las copias de *foto.jpg* en *floppy disks* son casos de un tipo distinto a las copias de *foto.jpg* en CD.

Finalmente, podríamos aclarar que lo que reúne en términos formales a todos estos tipos físicos es su posibilidad de ser interpretados en términos de 0 y 1 al interactuar con una computadora. Es a partir de este mecanismo de interpretación que, mediante distintas estructuras físicas, es posible generar un mismo código. Distintos soportes físicos inducen en la computadora la misma secuencia de voltajes de valor discreto y, así, de bits. En definitiva, debemos plantear dos tipos de materialidad formal: la del soporte y la del código. El tipo-código sería, ahora sí, múltiplemente realizable en distintos tipos-soporte.

Las propiedades ontológicas del *software*: copiabilidad y relacionalidad

Sobre la base del último apartado, podemos decir que, desde la propuesta de Blanco y Berti (2016), la ontología de un objeto digital admite copias, es decir, casos particulares de realización de un mismo tipo-código. De esta forma, podríamos tener muchas copias del archivo *foto.jpg* que serían idénticas en su tipo, no solo al interior de un tipo-soporte, sino de varios tipos-soporte distintos. Esta propiedad podría denominarse *copiabilidad*.

La segunda propiedad ontológica avanzada por estos autores es la *relacionalidad*, es decir, la dependencia del objeto digital respecto a un medio digital asociado. Por ejemplo, para que el archivo *foto.jpg* exista como tal, hace falta que interactúe con un medio digital capaz de interpretar archivos con formato *.jpg*, de otra forma, la cadena de bits que supone el objeto se mostraría como un archivo sin formato. En este sentido, es evidente que cualquier objeto digital depende de algún tipo de medio (por ejemplo, de un sistema operativo, de controladores, de códecs, etc.), por cuanto dicho medio es lo que permite interpretar, finalmente, la cadena de bits. Este es el motivo por el cual los autores consideran insuficiente el esquema bipartito de Kirschenbaum, ya que una misma materialidad formal (que se constituye en la cadena de bits) puede ser interpretada de distintas formas según el contexto digital, lo que modifica su estado ontológico (Blanco y Berti, 2016: 204). Una misma cadena de bits puede producir una secuencia cíclica de errores si es interpretada por un programa, o una canción, si es interpretada por otro.

La máquina universal

Para pensar el funcionamiento de las computadoras contemporáneas, Blanco y Berti acuden al concepto de máquina universal acuñado por Alan Turing en la primera mitad del siglo XX. Como señalan:

Una máquina de Turing puede ser definida a partir de un conjunto finito de estados, un alfabeto finito de símbolos de entrada y una función de transición, la que suele ser presentada como una tabla de transición. Los datos de entrada vienen en una cinta infinita (pero con una cantidad finita de símbolos no nulos), sobre la que habrá una posición diferenciada que indicará el símbolo corriente. Cada una de las filas de la tabla indica para cada estado y símbolo de entrada dados como se cambia el símbolo corriente de la cinta, si se mueve para la izquierda o la derecha la posición corriente, y cuál será el nuevo estado (Blanco y Berti, 2016: 209).

Basándose en este modelo, Turing planteó la existencia de una máquina universal (MUT, de aquí en adelante), capaz de tomar como entrada la codificación de cualquier otra máquina de Turing (MT, de aquí en adelante), llegando así a imitar su comportamiento. Esto habilita a la MUT a ejecutar cualquier programa, siempre y cuando se ponga a su disposición la codificación necesaria.

Blanco y Berti plantean que la versatilidad de las computadoras contemporáneas las exime del rótulo de mero artefacto técnico, para el cual hay un propósito definido en relación con intenciones previas. Al contrario, afirman que “esta versatilidad es una consecuencia directa de que pueden (sic) pensarse a las computadoras modernas como una encarnación particular de la llamada máquina universal” (Blanco y Berti, 2016: 199). De hecho, esta capacidad para emular cualquier tipo de comportamiento es lo que garantiza la realizabilidad múltiple de los objetos digitales en los dispositivos computacionales modernos, puesto que de ella depende la constitución del medio digital asociado y, con ello, la posibilidad de interpretación de todo código.

El problema de la realizabilidad múltiple del *software*: reproductibilidad física⁶

Basándonos en todo lo dicho, pondremos a prueba el esquema de análisis de Blanco y Berti (2016) a partir de un caso concreto. El escenario es el siguiente: tenemos

⁶ Agradezco a Fabio Espejo Mejía por ayudarme a elucidar los detalles técnicos y las problemáticas relativas al contraste entre modelos matemáticos y máquinas físicas/reales.

dos computadoras, ambas tienen instalado en su disco el mismo videojuego, que requiere un nivel elevado de recursos físicos para poder ejecutarse. El entorno digital en los dos dispositivos es el mismo en cualquier sentido relevante, ya que ambos pueden interpretar los archivos que componen al videojuego con el mismo formato y son lógicamente capaces de ejecutar las secuencias de código de la forma debida (tienen el mismo sistema operativo, programas-intérpretes, etc.). Ahora bien, las dos PC difieren en algo: una tiene más recursos físicos que la otra (en cuanto cantidad de memoria RAM, VRAM, potencia del CPU, etc.). Llamaremos *A* a la computadora con mayor cantidad de recursos y *B* a la computadora con menor cantidad.

Al darle doble clic al archivo ejecutable del videojuego en *A*, su código se ejecuta, junto con el contenido de todos los archivos que lo constituyen. Al hacer lo mismo en *B*, el videojuego emite un cartel de error a causa de que no existen recursos físicos suficientes para ejecutar el código. Detengámonos, por un momento, en esto. Según Blanco y Berti, ambas computadoras son encarnaciones de la MUT, ambas tienen el mismo medio digital asociado en un sentido relevante y ambas fueron sometidas a la interpretación de dos casos del mismo tipo-código. Sobre la base de los criterios de análisis mencionados previamente, el resultado de las dos operaciones debería ser similar. Sin embargo, podemos ver que no es así, ya que *A* puede ejecutar el programa mientras que *B*, no. ¿Qué es lo que ocurre y qué puede decirnos respecto a las teorías esbozadas previamente?

En principio, podríamos afirmar que se da un caso de falta de capacidad computacional. *B* no tiene espacio de almacenamiento suficiente en su memoria RAM/VRAM o capacidad de procesamiento para ejecutar el código demandado. Ahora bien, si *B* fuera una MUT, debería, por definición, ser capaz de emular el comportamiento de *cualquier* otra computadora, dado el caso de que se le entregue la codificación de la misma como entrada (cosa que, en este caso, ocurrió). Sin embargo, *B* no es capaz de ejecutar el programa por la sencilla razón de que no es una encarnación de la MUT. De hecho, *A* tampoco lo es, aunque haya logrado ejecutarlo. En este punto, cabe citar una reflexión de William Rapaport:

Pero, ¿es correcto limitar una computadora a un dispositivo físico? ¿No son las computadoras máquinas de Turing? ¿Deberíamos distinguir una computadora "real" de una abstracción matemática como una máquina de Turing? Pero, posiblemente, mi iMac, que seguramente es una computadora, si es que lo es, no es una máquina de Turing; más bien, puede ser modelado por una máquina de Turing (universal). (...) ¿qué pasa con la diferencia entre una computadora física real que solo puede calcular lo que sea prácticamente compu-

Sus aportes fueron fundamentales para el desarrollo de todo este escrito y, en especial, para este apartado.

table (es decir, sujeto a limitaciones razonables de espacio y tiempo) y una máquina de Turing abstracta y universal que no está así restringida? (Rapaport, 2018: 3)⁷.

Expliquemos esto un poco más. Para que una computadora realmente existente fuese un caso particular de la MUT, debería ser capaz de ejecutar cualquier código, no solo en un sentido lógico, sino en un sentido práctico. Todo proceso de cómputo requiere de una determinada capacidad de almacenamiento y procesamiento, y estos, a su vez, suponen una cierta disponibilidad de recursos físicos. Por esto mismo, una MUT, para ser tal, debería ser físicamente capaz de computar cualquier código imaginable, lo cual le requeriría tener recursos físicos infinitos. Es justamente la disparidad de recursos la que hizo la diferencia entre *A* y *B*.

Pensar las PC como realizaciones del modelo teórico MUT nos conduce a escenarios paradójicos como el precedente: una MUT que no puede ejecutar un código para el cual tiene las instrucciones apropiadas. La existencia de un nivel ontológico formal que se realiza en un nivel ontológico particular (la oposición entre materialidad formal y materialidad forense), en este caso, solo puede plantearse a costa de la reificación de un modelo teórico. Sin embargo, las computadoras reales no son modelos teóricos, sino entes físicos que tienen limitaciones físicas. Cualquier caso de realización de un objeto digital se da en un medio físico y, como tal, toda discusión relativa a su estado ontológico debe partir de un análisis que contemple sus condiciones físicas de existencia⁸.

La confusión multimedial

En el apartado anterior pusimos en juego el problema de la reproductibilidad física respecto al estado de la ontología digital, pero creemos que es necesario confrontar lo dicho previamente con una crítica posible, relativa al vínculo entre objeto digital y proceso multimedial. Para darle forma, nos apoyaremos en el análisis de Philippe Bootz (2011) sobre la poesía digital según el modelo procedimental de la comunicación.

Este autor propone entender el caso de la poesía digital programada desde un punto de vista comunicativo, que va desde el “polo autor” al “polo lector”. Según Bootz, “el autor fabrica un material, denominado fuente, destinado a un equipo tecnológico. Tiene como objetivo crear un acontecimiento gracias a este aparato

⁷ La traducción es nuestra.

⁸ Más adelante trataremos el problema general de la relación entre modelos y entidades físicas, así como varias cuestiones específicas de la relación entre las máquinas reales y el modelo de la MT.

en el dominio del lector utilizando la fuente. Este acontecimiento es denominado un “transitorio observable” (Bootz, 2011: 32). Podemos ver que se nos plantean tres escisiones fundamentales: la fuente, el equipo tecnológico y el transitorio observable. Estos tres elementos tienen, dentro del análisis de Bootz, una faz semiótica clave para el abordaje de la poesía digital en cuanto proceso comunicativo, pero a fines de nuestra exposición bastará con detenernos en la faz técnica. En términos concretos, la fuente es el programa y los datos (el código) que son compuestos en el dispositivo técnico al alcance del autor y reproducidos por el dispositivo técnico al alcance del lector; el equipo es el conjunto de los dispositivos técnicos del lector y del autor, y lo que lo vincula (las PC, internet, los controladores de ambas PC, etc.); el transitorio observable es el “acontecimiento multimedia producido en la ejecución del programa (...) es del orden de la señal. Puede ser grabada y tratada fuera de la lectura” (Bootz, 2011: 32-33).

¿En qué sentido el esquema de Bootz podría utilizarse para formular una crítica a nuestro ejemplo del videojuego? Para esclarecerlo, hace falta plantear la relación entre los tres ámbitos propuestos por el autor y la ontología del objeto digital según la formulamos en este texto. Habíamos señalado que un objeto digital, según Blanco y Berti (2016), cuenta con dos propiedades fundamentales: copiabilidad y relacionalidad. Podríamos decir que la copiabilidad está fuertemente vinculada con el ámbito de la fuente, mientras que la relacionalidad lo está con el del equipo tecnológico. Esto es así porque lo que se copia es la fuente, es decir, el código (la cadena de bits) y porque el medio digital asociado está contenido dentro del equipo tecnológico. Pero notemos que ninguna de las dos propiedades del objeto digital se vincula con el tercer ámbito del esquema de Bootz: el transitorio observable. Le alcanzaría a un crítico nuestro, entonces, con demostrar que el problema de la reproductibilidad física afecta solo al ámbito del transitorio observable para demostrar que no es un problema relativo a la ontología del objeto digital. Vemos, entonces, por qué es necesario atender a esta crítica: si es acertada, nuestro caso de análisis se volvería inadecuado puesto que atendería, más bien, al ámbito de la reproducción multimedial o de la experiencia subjetiva de encuentro con ella (algo que quizás interesaría más a fenomenólogos que a ontólogos).

Revisemos, pues, nuestro ejemplo bajo el esquema de análisis de Bootz a fin de ver qué tan acertada sería esta crítica. Habíamos planteado un escenario en que un código es copiado en dos PC que tienen el mismo medio digital asociado, pero distintos recursos físicos (una tiene menos recursos que la otra); la diferencia de recursos causa, entonces, que la que tiene más lo ejecute sin problemas y que la que tiene menos dé un cartel de error. Ahora, acudiendo al esquema de Bootz, debemos situar el problema de reproducción en alguno de los ámbitos de la tríada: fuente, equipo tecnológico, transitorio observable, pero ¿en cuál? En principio, podríamos descartar que se dé en la fuente, puesto que el código/programa es el mismo en ambas PC. Restan, entonces, el equipo tecnológico y el

transitorio observable. Dada la dependencia del transitorio observable respecto a las condiciones de funcionamiento del equipo tecnológico, un problema que se dé en el último debe afectar, también, al primero. Ahora bien, ¿ocurre lo mismo a la inversa? Pareciera que no, ya que el acontecimiento multimedia puede ser grabado y escindido del proceso computacional que le da origen en el caso de la reproducción del videojuego.

Es concebible una versión extendida de nuestro escenario problemático en la que el transitorio observable de la PC que sí puede reproducir el videojuego es capturado por *software* (se registra el audio y video de una ronda de juego) y enviado a la PC que dio error en forma de archivo de video. La PC que fue incapaz de reproducir el videojuego será capaz de reproducir el archivo de video en cuestión, demostrando que sus problemas relativos al transitorio observable son contingentes, mientras que sus problemas relativos a la reproducción del juego como programa no lo son. Esto último es importante ya que demuestra que, en nuestro caso de análisis, el problema de la reproductibilidad física afecta al ámbito del transitorio observable solo en un nivel secundario y a consecuencia de un problema en el equipo tecnológico⁹.

⁹ Se nos podría señalar (como sugirió un revisor anónimo de este artículo) que un agente que pretenda hacer uso de ambos dispositivos notaría una diferencia en los outputs de manera muy rápida: la reproducción en video no soporta contrafácticos, es decir, no tolera las mismas acciones que un usuario o agente podría haber efectuado en el sistema que puede ejecutar correctamente la performance instruida por el código. Así planteada, esta diferencia entre los escenarios de uso atendería al transitorio observable, por lo que habría un argumento nuevo para afirmar que el problema tratado pertenece únicamente a dicho nivel. Esta crítica señala una cuestión interesante: los escenarios planteados difieren en términos de sus posibilidades de interacción. Si bien concordamos con que existe tal diferencia (y eso es lo que permite distinguir el video del videojuego), no creemos que pertenezca al nivel del transitorio observable. La diferencia que dicho agente podría notar no estaría en los outputs (si por ellos entendemos a los outputs multimediales), sino que se volvería evidente a partir de ellos. La interactividad presupone la posibilidad de encuentro entre el dispositivo computacional y un conjunto de variables no-locales, es decir, del entorno (para un tratamiento más exhaustivo del concepto de “interacción” en el marco de la ciencia computacional, cfr. Wegner, 1997). Ni la PC en cuanto dispositivo computacional ni dichas variables (las acciones del usuario, en este caso) son algo que se pueda capturar mediante una grabación del acontecimiento multimedia. Dado que el transitorio observable es del orden de la señal y que, por definición, puede ser grabado y tratado fuera del acontecimiento, es evidente, entonces, que las acciones posibles para un agente o usuario durante su desarrollo no pueden formar parte de él. Por ello, no solo sería falso afirmar que el transitorio observable difiere entre la ronda de juego y la reproducción de video (el output es el mismo), sino que es justamente el hecho de que no difiere (frente a tal o cual acción del usuario) lo que permite distinguir ambos escenarios en términos de interactividad.

El esquema de Bootz nos permite, sabiendo esto, hacer una crítica a las propiedades del objeto digital según son entendidas por Blanco y Berti (2016). En particular, nos habilita a revisar la noción de relacionalidad, puesto que, al definirla en términos de relación con un medio digital, y al entender lo digital solo como un vínculo entre código e intérpretes, se pierde de vista la dimensión de relacionalidad física en toda su extensión. Bootz mismo nos ofrece una categoría particular para analizar nuestro caso problemático, la *labilidad técnica*:

El contexto técnico está dado por el conjunto de las diferencias de parámetros entre la máquina del autor y la del lector. Entonces, determina en parte el transitorio observable. De este modo, una fuente dada puede producir la ejecución de transitorios observables diferentes según el contexto tecnológico. Este fenómeno es conocido en el modelo con el nombre de “labilidad técnica” (Bootz, 2011: 33).

En nuestro caso no hay una relación uno a uno entre persona que crea el juego y persona que juega, pero el criterio aplica (podríamos pensar la relación entre las máquinas de los programadores del videojuego y las de los jugadores). Vemos, entonces, dónde ocurre el problema en nuestro ejemplo: en la diferencia de parámetros entre las máquinas, lo que induce un caso dramático de labilidad técnica. Ahora bien, la palabra “parámetros”, tal como está usada en la última cita, es tan abstracta que no se ve cómo esta definición podría ayudar a nuestra postura más que a la de Blanco y Berti, pero el mismo Bootz nos da la respuesta cuando, dentro del equipo tecnológico, incluye *todo* lo que hace a las PC y su relación en cuanto dispositivos técnicos, y no solo la relación objeto-medio digital, entendidos como código e intérprete (en la que entran los controladores, el sistema operativo, etc.). La noción de “equipo tecnológico” nos permite contemplar, entre otras cosas, la cantidad de memoria RAM, VRAM, potencia del CPU y demás como partes del ensamblado, mientras que la noción de “medio digital asociado” no lo permite¹⁰.

Modelos y cajas negras

Resta abordar un último tema antes de formular nuestra propuesta: la relación general entre entidades físicas y modelos. Los modelos tienen una importancia

¹⁰ Esto es así por lo que comentamos previamente: lo digital, en términos de Blanco y Berti (2016), está pensado en relación con la MUT como modelo de intérprete. En dicho modelo la disponibilidad de recursos físicos como “capacidad de almacenamiento” no es un factor a considerar porque se asume una capacidad irrestricta de cómputo (cinta y tiempo ilimitados).

central en muchos contextos científicos¹¹. La centralidad de los modelos inflacionarios en cosmología o de los evolutivos y de doble hélice del ADN en biología son buenos ejemplos de ello. Los modelos plantean preguntas serias en semántica (¿cómo representan, si es que lo hacen?), ontología (¿qué tipo de cosas son?), epistemología (¿cómo aprendemos y explicamos con ellos?) y, por supuesto, en otros dominios dentro de la filosofía de la ciencia. Un desarrollo pormenorizado de todas estas cuestiones excede por mucho las posibilidades de este texto; por ello, trataremos un tema en particular: su función epistemológica. Creemos que esto es necesario para determinar si la MUT podría considerarse un modelo apropiado de las computadoras contemporáneas. Si este fuera el caso, podría afirmarse que, si bien hay una diferencia clara entre ambas, las últimas pueden ser modeladas por la primera.

¿Qué hace, en este sentido, que un modelo sea apropiado? Una de las principales razones por las que los modelos juegan un papel tan importante en la ciencia es que realizan una serie de funciones cognitivas. Por ejemplo, ofician de vehículos para aprender sobre el mundo. Una buena parte de la investigación científica se lleva a cabo sobre modelos en vez de sobre la realidad misma porque al estudiarlos podemos descubrir características y determinar hechos sobre los sistemas que representan: los modelos permiten el “razonamiento sustitutivo” (Swoyer, 1991).

Una vez que tenemos conocimiento sobre el modelo, este tiene que ser “traducido” en conocimiento sobre el sistema de destino (la entidad física modelada). Es en este punto que la función representacional de los modelos se vuelve importante: si un modelo representa, entonces puede instruirnos sobre la realidad porque (al menos algunas de) sus partes o aspectos tienen sus correlatos en el mundo. Pero si dicha instrucción está conectada con la representación y si hay diferentes tipos de representaciones (analogías, idealizaciones, etc.), entonces también hay diferentes maneras en las que un modelo puede instruirnos. Pero, ¿cómo pueden cumplir los modelos una función explicativa, dado que típicamente involucran idealizaciones? ¿Estos modelos explican a pesar de o debido a las idealizaciones que implican? ¿Y qué tipo de explicación proporcionan?

Existe una larga tradición que exige que los *explanans* de una explicación científica sean verdaderos. Encontramos este requisito en el modelo nomológico-deductivo (Hempel, 1965), así como en la literatura más reciente. Por ejemplo, Strevens afirma que “ninguna descripción causal de la explicación (...) permite que los modelos no verídicos expliquen” (Strevens, 2008: 297)¹². Los autores que trabajan en esta tradición niegan que las idealizaciones hagan una contribución

¹¹ En este apartado tomaremos como base a Frigg y Hartmann (2020) para el desarrollo general de la problemática.

¹² La traducción es nuestra.

positiva al proceso explicativo y exploran cómo los modelos pueden explicar a pesar de ser idealizados. McMullin (1968; 1985) argumenta que una explicación causal basada en un modelo idealizado omite solo las características que son irrelevantes para la tarea explicativa respectiva. Strevens (2008) señala que un modelo causal explicativo debe proporcionar una representación precisa de las relaciones o procesos causales relevantes que este comparte con el sistema de destino. Los supuestos idealizados de un modelo no marcan una diferencia para el fenómeno bajo consideración y, por lo tanto, son explicativamente irrelevantes.

Consideramos que la exigencia de verdad para el *explanans* de una explicación científica es válida. También aceptamos el uso de idealizaciones, siempre y cuando se las pueda justificar sobre la base de las condiciones especificadas. Podría plantearse, entonces, que el recurso a la MUT es explicativamente apropiado si se la asume como un modelo con dichas características. Ahora bien, ¿es ese el caso? Creemos que no. En principio, está claro que la disponibilidad de memoria y tiempo ilimitados del modelo no es verídica, en el sentido de que no es una condición que pueda darse en una máquina real. Sin embargo, podría argumentarse que la idealización implicada por el modelo elimina factores explicativos irrelevantes y acentúa otros factores relevantes. Por ejemplo:

1. La relación “lógica” entre entradas y salidas de cualquier computadora real puede simularse por una MUT programada para imitar esa máquina particular, a condición de que esta última sea vista como una “caja negra” (atendiendo solo a sus *inputs* y *outputs*).
2. De lo anterior se sigue que una declaración sobre las limitaciones de las MT también se aplicará a las computadoras reales.
3. Existe un límite para la memoria que posee cualquier máquina actual, pero este límite puede aumentar arbitrariamente con el tiempo. Las MT nos permiten hacer afirmaciones sobre algoritmos que (teóricamente) se mantendrán para siempre, independientemente de los avances en la arquitectura de las máquinas informáticas convencionales.

Si bien estos señalamientos son útiles para tratar los límites teóricos de la computación, surgen varios problemas cuando intentamos aplicarlos a la computación real. Respecto a 1, podríamos decir que es imposible determinar qué sistema computacional es realmente un individuo o máquina testeada si lo asumimos como una caja negra, puesto que un mismo comportamiento puede ser explicado por (potencialmente infinitos) modelos compatibles. Las capacidades de entrada-salida compatibles con una MT particular revelada en cualquier secuencia finita y arbitrariamente larga de pruebas podrían, en principio, ser producidas por infinitas MT diferentes o incluso por una máquina mucho más limitada que una MT (por ejemplo, una que recurra a una tabla de entrada y salida fija y no sea

capaz de producir ninguna respuesta para una entrada que no esté en su tabla)¹³. Esta limitación del modelo de caja negra podrá ser irrelevante en ciertos contextos técnicos, pero no en un debate filosófico (en particular, si estamos tratando de determinar la ontología del objeto estudiado).

Respecto a 2 y 3, cabe señalar que “es posible que algunas funciones computables de Turing *nunca* sean computables en la práctica, ya que pueden requerir más memoria de la que se puede construir utilizando toda la cantidad (finita) de átomos en el universo” (De Mol, 2021)¹⁴. Esta limitación difícilmente pueda considerarse irrelevante, ya que establece constricciones físicas irrevocables (y, en ese sentido, nos dice tanto de los límites de la computación como del modelo de la MT). Además, elucidar qué funciones computables de Turing son físicamente imposibles de computar es un objetivo central de la teoría de la complejidad y un factor definitorio para el tratamiento de ciertos problemas computacionales en términos de la posibilidad o imposibilidad de implementar algoritmos que los resuelvan. Por ejemplo, algunos sistemas de encriptado se fundamentan en que su resolución no es físicamente computable.

Finalmente, el estudio de ciertos contextos reales de computación requiere tener en cuenta las limitaciones físicas de recursos porque estas juegan un rol causal en el marco de la ontología relacional de los objetos digitales. Casos como el de nuestro escenario problemático serían imposibles de explicar acudiendo únicamente al modelo de la MUT. Estos motivos son suficientes, a nuestro juicio, para negar que el modelo en cuestión sea explicativamente apropiado, ya que las condiciones no verdícas (idealizadas) que implica eliminan factores explicativamente relevantes.

Nuestra propuesta ontológica: identidad físico-digital

En primer lugar, afirmamos que el único criterio válido para predicar identidad entre dos objetos digitales es el reconocimiento de una total identidad entre sus atributos físicos. Esto, por el simple hecho de que los objetos digitales son un subtipo dentro del tipo objetos-físicos, cuya ontología se establece relacionamente no en el marco de un medio digital asociado, sino en el de un equipo tecnológico (más amplio y físico). Es decir que, si bien compartimos el espíritu de crítica al dualismo digital que proponen Blanco y Berti (2016), nuestra posición es más radical: afirmamos que dicho dualismo (incluso en la separación *hardware/software*) existe solo por abordar las computadoras reales desde un modelo inapropiado. Afirmar esto implica posicionar nuestra teoría ontológica de lo digital

¹³ Para una defensa ampliada de esta misma tesis, cfr. Sloman (2016).

¹⁴ La traducción y las cursivas son nuestras.

en el marco de un monismo fiscalista: “digital” es solo el nombre que damos a ciertos fenómenos físicos¹⁵.

Alguien podría señalar en este punto que, como se predica en la definición de materialidad forense, no existen dos objetos físicos exactamente idénticos (si tomamos como parte de sus atributos la ubicación espacio-temporal), por lo que una modelización apropiada de un objeto digital específico sería imposible. Frente a esto se podrían proponer dos respuestas: una, que habría que discutir si la ubicación espacio-temporal es un atributo *del* objeto físico (¿dos moléculas de H₂O no podrían considerarse como casos espacio-temporalmente discernibles distintos de un mismo tipo físico?); otra, que eso no supone un problema para nuestra teoría de la identidad. Estamos dispuestos a admitir la posibilidad de que, de hecho, no existan dos objetos digitales absolutamente idénticos, puesto que, si se lo pudiera admitir para moléculas, ¿por qué no para copias de un archivo? Además, los objetos digitales, entendidos en el marco de procesos computacionales físicos, se dan en un plano macroscópico (involucran gran parte del aparataje de una computadora real); dicha complejidad ontológica dificulta la posibilidad de encontrar dos casos físicamente idénticos (cosa más sencilla para entidades microscópicas como una molécula). Al contrario, suponer una falta de identidad en estos casos es lo que nos permite abordar fenómenos relevantes del ámbito digital que el análisis basado en el modelo de Turing elude.

Con lo dicho ya estamos en condiciones de abordar la siguiente pregunta: ¿cómo es posible hablar de “copias de archivo” en el marco de una teoría de la identidad físico-digital? La respuesta es: de la misma manera en que es posible hablar de copias impresas de un mismo texto. Las copias de archivo son objetos ontológicamente distintos (en un sentido físico), pero que, dependiendo del modelo con que se los compare, pueden considerarse funcionalmente idénticos. Su condición ontológica-física les es intrínseca y su condición funcional es parcialmente arbitraria¹⁶. Así, podríamos decir que dos copias de foto.jpg ubicadas en un disco rígido se parecen más entre sí, ontológicamente hablando (comparten más atributos físicos), que lo que se parecen respecto a una copia de foto.jpg ubicada en un CD (aunque funcionalmente pueda considerarse idénticas a las tres). Esta gradualidad de las similitudes o diferencias entre distintas copias se manifiesta, a su vez, en similitudes o diferencias graduales en su realización como procesos de cómputo, lo que prueba nuestro punto: el objeto digital es *idéntico* al proceso físico.

¹⁵ Una forma simplificada e intuitiva de expresar lo dicho hasta aquí sería: estamos más cerca de considerar los objetos digitales como configuraciones materiales de voltajes que como entidades abstractas hechas de ceros y unos.

¹⁶ Recuérdense el problema mencionado sobre la imposibilidad de determinar qué modelo es el que explica el funcionamiento real de una caja negra.

Conclusión

En este trabajo realizamos un análisis de la posición de Blanco y Berti (2016) en relación con la ontología de los objetos digitales y su contraste con otras posiciones que ellos critican. Abordamos las propiedades ontológicas avanzadas por ellos (copiabilidad y relacionalidad) y las implicaciones que estas tienen respecto a una teoría de la realizabilidad múltiple. Explicamos el vínculo entre estas propiedades y el concepto de máquina universal de Alan Turing, que los autores consideran “encarnado” en las PC contemporáneas. A continuación, postulamos un escenario problemático que funciona como contraejemplo para la teoría de la realizabilidad múltiple del *software*, dado que se revela insuficiente como insumo para la explicación. Luego, dimos una interpretación propia del escenario planteado en términos de un problema de reproductibilidad física. Más adelante, abordamos y respondimos posibles críticas a nuestro ejemplo basadas en la relación entre objeto digital y *output* multimedial, a partir del modelo procedimental de la comunicación de Bootz (2011), y de problemas relativos a la relación entre modelos y entidades físicas. Finalmente, propusimos una comprensión de la ontología de los objetos digitales en términos de identidad físico-digital y de relacionalidad situada en el marco de un equipo tecnológico.

Bibliografía

Fuentes

Bickle, John (2020), “Multiple realizability”, en Zalta, Edward Nouri (ed.), *The Stanford encyclopedia of philosophy*, California, Metaphysics Research Lab, Stanford University, [disponible en <https://plato.stanford.edu/archives/sum2020/entries/multiple-realizability/> - consultado el 27 de septiembre de 2021].

Blanco, Javier y Berti, Agustín (2016), “No hay hardware sin software: crítica del dualismo digital”, *Quadranti*, vol. 4, n° 1-2, pp. 197-214, [disponible en <https://ri.conicet.gov.ar/handle/11336/96809>].

Bootz, Philippe (2011), “La poesía digital programada: una poesía del dispositivo”, en Kozak, Claudia (comp.), *Poéticas tecnológicas, transdisciplina y sociedad. Actas del Seminario Internacional Ludión/Paragraphe*, Buenos Aires, Exploratorio Ludión, pp. 31-40.

De Mol, Liesbeth (2021), “Turing machines”, en Zalta, Edward Nouri (ed.), *The Stanford encyclopedia of philosophy*, California, Metaphysics Research Lab, Stan-

ford University, [disponible en <https://plato.stanford.edu/entries/turing-machine/> - consultado el 25 de mayo de 2022].

Frigg, Roman y Hartmann, Stephan (2020), "Models in science", en Zalta, Edward Nouri (ed.), *The Stanford encyclopedia of philosophy*, California, Metaphysics Research Lab, Stanford University, [disponible en <https://plato.stanford.edu/archives/spr2020/entries/models-science/> - consultado el 25 de mayo de 2022].

Hempel, Carl Gustav (1965), *Aspects of scientific explanation and other essays in the philosophy of science*, Nueva York, Free Press.

Kirschenbaum, Matthew Gary (2012), *Mechanisms: New media and the forensic imagination*, Cambridge, The MIT Press.

McMullin, Ernan (1968), "What do physical models tell us?", en van Rootselaar, Bob y Staal, Johan Frederik (eds.), *Methodology and philosophy of science III (Studies in logic and the foundations of mathematics 52)*, Amsterdam, North-Holland, pp. 385-396.

----- (1985), "Galilean idealization", *Studies in history and philosophy of science Part A*, vol. 16, n° 3, pp 247-273.

Putnam, Hilary (1960), "Minds and machines", en Hook, Sidney (ed.), *Dimensions of mind*, Londres, Collier-Macmillan, pp. 138-164.

Rapaport, William Joseph (2018), "What is a computer? A survey", *Minds and machines*, vol. 28, n° 3, pp. 385-426, [disponible en <https://doi.org/10.1007/s11023-018-9465-6>].

Strevens, Michael (2008), *Depth: An account of scientific explanation*, Londres, Harvard University Press.

Swoyer, Chris (1991), "Structural representation and surrogative reasoning", *Synthese*, vol. 87, n° 3, pp. 449-508.

Bibliografía referida

Robinson, Howard (2020), "Dualism", en Zalta, Edward Nouri (ed.), *The Stanford encyclopedia of philosophy*, California, Metaphysics Research Lab, Stanford University, [disponible en <https://plato.stanford.edu/archives/fall2020/entries/dualism/> - consultado el 27 de septiembre de 2021].

Sedgewick, Robert y Wayne, Kevin (2011), *Algorithms*, Nueva Jersey, Addison-Wesley Professional.

Sloman, Aaron (2016), *Judging chatbots without opening them: Limitations of "black-box" tests*, Birmingham, University of Birmingham, [disponible en <https://www.cs.bham.ac.uk/research/projects/cogaff/misc/black-box-tests.pdf> - consultado el 25 de mayo de 2022].

Wegner, Peter (1997), "Why interaction is more powerful than algorithms", *Communications of the ACM*, vol. 40, n° 5, pp. 80-91.

Wetzel, Linda (2018), "Types and tokens", en Zalta, Edward Nouri (ed.), *The Stanford encyclopedia of philosophy*, California, Metaphysics Research Lab, Stanford University, [disponible en <https://plato.stanford.edu/archives/fall2018/entries/types-tokens/> - consultado el 25 de mayo de 2022].