

Hacia una Ontología del Videojuego: Análisis de la Teoría de Ian Bogost desde la Ontología
Serial del Software

Heimar Eduardo Mendoza Cely

Trabajo de Grado para Optar por el Título de Magíster en Filosofía

Director

Jorge Francisco Maldonado Serrano

Doctor en filosofía

Universidad Industrial de Santander

Facultad de Ciencias Humanas

Escuela de Filosofía

Bucaramanga

2024

Dedicatoria

A mi familia, que cree incondicionalmente en mí y me apoya en la persecución de mis sueños. Les dedico este trabajo como una de las tantas victorias venideras.

A Mechas, mi fiel acompañante felina que recientemente nos dejó atrás. Te deseo los sueños más dulces.

Agradecimientos

Agradezco profundamente a la Universidad Industrial de Santander en general, y de manera especial a la Escuela de Filosofía, por brindarme la invaluable oportunidad de continuar con mi formación académica. Este trabajo es producto del estímulo intelectual de su cuerpo docente, en particular de mi director de tesis, el Dr. Jorge Francisco Maldonado Serrano, quien nunca dudó de mis habilidades.

Agradezco a mis compañeros de cohorte por acogerme y brindarme su cálida amistad. La cooperación y el compañerismo son el sello distintivo de la XIV cohorte de la maestría en filosofía. En este sentido, este proyecto no termina conmigo. Ansío con emoción y regocijo la graduación de aquellos compañeros que, temporalmente, se han quedado atrás. Aunque nuestros caminos divergen momentáneamente, sé que todos seremos magísteres en filosofía, unidos por el lazo del compromiso con el conocimiento y de la pasión por la filosofía.

Tabla de contenido

	Pág.
Introducción.....	8
1. La Ontología Serial del Software como Estructura para el Análisis.....	11
1.1 ¿Por qué la Ontología Serial del Software?.....	11
1.1.1 El software es más que código.....	11
1.1.2 La Flexibilidad de la Ontología Serial del Software.....	12
1.1.3 Comprender lo digital y sus funcionalidades.....	15
1.2 Aspectos Conceptuales.....	19
2. Una Aproximación a la Propuesta Teórica de Bogost.....	26
2.1 Primer Método de Análisis: Estudios de Plataformas.....	26
2.1.1 Atari VCS: Trazos sobre Qué Es un Videojuego.....	27
2.1.2 El Proceso Creativo en la Atari VCS.....	30
2.2 Segundo método de análisis: la proceduralidad.....	32
2.2.1 Operación Unitaria.....	33
2.2.2 Análisis Unitario.....	34
2.2.3 Crítica Comparativa.....	36
2.2.4 Juegos Persuasivos.....	37
2.2.4.1 Juegos Persuasivos: Presupuestos Conceptuales.....	38
2.2.4.1.1 Proceduralidad.....	39
2.2.4.1.2 Retórica.....	43
2.2.4.2 Juegos Persuasivos: la Retórica Procedural en la Práctica.....	44
2.2.4.2.1 El Acto de Jugar.....	45

2.2.4.2.2 El Rol del Jugador: Completar los Entimemas Procedurales.....	48
2.3 Aspectos generales de la teoría de Bogost.....	52
3. La teoría de Bogost a la luz de la Ontología Serial del Software	53
3.1 Revisión de los dos métodos de análisis.....	53
3.2 La especificidad de los videojuegos.....	63
4. Conclusiones	72
Referencias Bibliográficas.....	76

Resumen

Título: Hacia una Ontología del Videojuego: Análisis de la Teoría de Ian Bogost desde la Ontología Serial del Software*

Autor: Heimar Eduardo Mendoza Cely**

Palabras clave: digital, videojuegos, ontología serial del software, plataformas, proceduralidad, Bogost.

Descripción: en el presente trabajo se propone llevar a cabo un análisis de la obra del filósofo Ian Bogost a partir de la Ontología Serial del Software como estructura para el análisis. Para la Ontología Serial del Software el estudio filosófico del software se revela como fundamental para comprender la relación entre el ser humano y lo digital. Se adopta una posición crítica frente a la noción de virtualidad, mientras se favorece la concepción de lo digital, lo que permite desprenderse de los prejuicios asociados a lo virtual. El software, y por extensión, los videojuegos, abarcan tanto elementos simbólicos como materiales. No obstante, al considerar el software como un conjunto de múltiples series en funcionamiento, se reconoce que este no es una entidad estática, sino más bien un proceso. La Ontología Serial del Software, por lo tanto, se configura como una ontología procesal. Se considera que la estructura teórica de esta propuesta resulta apropiada para abordar la teoría de Ian Bogost sobre los videojuegos.

* Trabajo de grado

** Facultad de Ciencias Humanas. Escuela de Filosofía. Maestría en Filosofía. Director: Jorge Francisco Maldonado Serrano. Doctor en Filosofía.

Abstract

Title: Towards and ontology of the videogame: an analysis of Ian Bogost's theory from the Serial Ontology of Software*

Author: Heimar Eduardo Mendoza Cely**

Key words: digital, videogames, serial ontology of software, platforms, procedurality, Bogost.

Description: This work proposes to conduct an analysis of the philosopher Ian Bogost's work using the Serial Ontology of Software as a framework for analysis. For the Serial Ontology of Software, the philosophical study of software is deemed essential for understanding the relationship between humans and the digital. A critical stance is taken towards the notion of virtuality in favor of the conceptualization of the digital, thereby shedding the biases associated with the virtual. Software, and by extension, video games, encompass both symbolic and material elements. However, by considering software as a set of multiple ongoing series, it is acknowledged that it is not a static entity but rather a process. The Serial Ontology of Software, therefore, is configured as a procedural ontology. The theoretical structure of this proposal is deemed suitable for addressing Ian Bogost's theory on video games.

* Degree work

** Faculty of Humanities. School of Philosophy. Master's in philosophy. Director: Jorge Francisco Maldonado. Doctor in Philosophy.

Introducción

El interés académico por los videojuegos no ha hecho más que incrementar con el paso de los años. La variedad de propuestas de estudio van desde intentos por comprender su fundamento ontológico según el tipo de pensamiento que opera en el jugador (Cardero et al., 2014), o sus aportes a la narratología en cuanto a la presentación y experiencia de historias (Collantes, 2013), o su impacto sociopolítico en la conformación y estructuración de sociedades *in-game* (Grimes & Feenberg, 2009). Para la filosofía, los videojuegos también se han consolidado como un objeto de estudio, en particular para Ian Bogost, un reconocido académico dentro de los estudios anglosajones sobre videojuegos. Bogost ha abordado, entre otras cosas, el debate entre narratología y ludología (Bogost, 2008c) y la labor del crítico de videojuegos (Bogost, 2008c, 2015). Sin embargo, desarrollar una ontología del videojuego no ha formado parte de sus intereses académicos: «Instead, I would rather leave the work of building ontologies and typologies to the many capable theorists who are already undertaking such projects.» (2008c, p. xiii). Con todo, hay en los aportes de Bogost una visión particular sobre los videojuegos como artefactos con capacidad expresiva que deja entrever una ontología del videojuego. En este sentido, en el presente trabajo se plantea explicitar esta ontología del videojuego por medio de una valoración de la obra de Bogost. Para llevar a cabo esta tarea se recurre tanto al análisis textual y conceptual de la obra de Bogost como a su análisis por medio de la Ontología Serial del Software.

La Ontología Serial del Software es una propuesta de lectura de la tecnología a partir de la filosofía. Se trata de una visión ontológica que comprende al software desde cuatro series, a saber, código, flujo de energía, circuitos e intencionalidad, y desde dos dimensiones del software, a saber, la dimensión serial y la dimensión estructural. En la dimensión serial, estas series son distinguibles

e inteligibles ontológicamente, pero desde la dimensión estructural, convergen unas con otras lógicamente, es decir, funcionan al unísono (Maldonado et al., 2020). En este sentido, la dimensión serial entiende al software como un objeto fijo, mientras que la dimensión estructural lo entiende como proceso en funcionamiento. Ambas dimensiones forman parte de la Ontología Serial del Software, luego, si bien son distinguibles, a nivel teórico se refieren a lo mismo. Las series del software son ontológicamente distinguibles en su dimensión serial, pero, en tanto que, una vez en funcionamiento se integran entre sí, las series están vinculadas en su dimensión estructural.

Para llevar a cabo el análisis propuesto, se ha estructurado el trabajo en cuatro partes. En primer lugar, se hace una revisión de la Ontología Serial del Software como método para el análisis. Sus alcances, sus principales características y sus ventajas. En segundo lugar, se hace una revisión general de la teoría de Ian Bogost. La idea de esta sección es sugerir una forma y una estructura a la teoría de Bogost a modo de preparación para su posterior análisis. Se trata de un análisis textual y conceptual de la teoría de Ian Bogost, señalando sus principales argumentos, sus conceptos y sus aportes más destacables. En tercer lugar, se lleva a cabo el análisis de la teoría de Bogost a partir de la Ontología Serial del Software, según lo presentado en el primer y segundo capítulo. Una última sección contiene las conclusiones del trabajo.

Se ha decidido seguir esta estructura en orden de facilitar el análisis. Como teórico de los videojuegos, la producción literaria de Bogost es muy vasta y variada, por lo que revisarla en su totalidad resulta complicado. Además, se considera que las obras seleccionadas (*Racing the Beam* (2009), *Unit Operations* (2008c) y *Persuasive Games* (2007)) contienen la parte más sustanciosa de su teoría, al menos para los objetivos del presente trabajo. De esta manera, el segundo capítulo se ocupa de una revisión general de la teoría de Bogost previa al análisis que se lleva a cabo en el

capítulo tres. Con esto en mente, ya se puede señalar hacia dónde se quiere llegar: explicitar en la teoría de Bogost una ontología del videojuego, con la cual se entendería con mayor precisión algunas de sus afirmaciones y argumentos.

El trabajo cuenta, pues, con un doble sentido, a saber, es de carácter prescriptivo y descriptivo al mismo tiempo. Su carácter prescriptivo consiste en la revisión analítica de la teoría de Bogost por medio del análisis textual y conceptual, por un lado, y del análisis mediante la Ontología Serial del Software, por el otro. Se prescribe una manera de analizar la teoría de Bogost. Su carácter descriptivo consiste en comprender qué es un videojuego para Bogost, bajo los términos de la Ontología Serial del Software. Se intenta llegar a una descripción del videojuego en un filósofo en particular, y con ello, aportar a los estudios ontológicos sobre videojuegos.

1. La Ontología Serial del Software como Estructura para el Análisis

1.1 ¿Por qué la Ontología Serial del Software?

Las posibilidades que ofrece la Ontología Serial del Software como estructura para el análisis se pueden determinar a partir de tres elementos. Primero, ofrece una interpretación del software que toma distancia de la agotada y poco precisa distinción entre hardware y software, o lo que es lo mismo, de la distinción entre lo físico y lo metafísico. Segundo, es susceptible a la adición de nuevas series, lo que implica cierta flexibilidad teórica. Tercero, la Ontología Serial del Software se construye como una apuesta para comprender la relación entre la humanidad y lo digital, o más precisamente, entre la humanidad y el universo digital. En lo que sigue se revisará cada cuestión por separado.

1.1.1 El software es más que código

En primer lugar, el software no es solo código, pero tampoco es algo distinto y ajeno a él. Al contrario, el software es código, pero solo en parte. El código, por sí solo, no logra nada, se requiere de un sustrato material que reciba sus instrucciones. En otras palabras: «it is not correct to assume that codification is independent of the hardware for which it is intended.» (Maldonado et al., 2020, p. 122). En este sentido, el software implica tanto el sustrato material como el código que permite a ese sustrato funcionar. Eso es lo que significa, precisamente, que una pieza de

hardware esté codificada, a saber, que posee un código incorporado en sí mismo que le permite funcionar.

Por otro lado, el vínculo ontológico entre el código y el hardware que codifica se desenvuelve en el funcionamiento. De hecho, desde la dimensión estructural del software, se entiende que el software en funcionamiento *es*, propiamente, el software: «the kind of *thing* that can only exist or is accepted as existent if various processes and various components function» (Maldonado et al., 2020, p. 123). El software funciona (existe) como si se tratase de una expresión en sentido spinoziano. Para que el hardware funcione se necesita de un código en ejecución, y este último necesita, a su vez, de un sustrato material, lo que comprende todos los componentes físicos, es decir, circuitos y flujo de energía que le permitan llevar a cabo su codificación, es decir, su transmisión de indicaciones. Así pues, en primera medida se argumenta que la existencia del software, es decir, el software en su dimensión estructural implica una alta complejidad que sobrepasa el carácter de mero código. En la siguiente sección se revisará la flexibilidad teórica que subyace a esta propuesta.

1.1.2 La Flexibilidad de la Ontología Serial del Software

En segundo lugar, desde la dimensión serial las series del software son, al igual que los atributos de la Sustancia spinoziana (Spinoza, 2014, E1P10), independientes entre sí a nivel ontológico. Para Spinoza cada atributo expresa la esencia de la Sustancia de una determinada manera. Pero, en tanto expresiones de la misma Sustancia, los atributos se encuentran vinculados. Los atributos son independientes ontológicamente, pero se encuentran vinculados entre sí en tanto atributos de una misma Sustancia. Esta misma argumentación se aplica a las series, de modo que

cada una expresa al software de cierta manera, es decir, cada una es una perspectiva distinta del software.

En la teoría spinoziana los atributos del pensamiento y de la extensión son cognoscibles al ser humano dado que él mismo es pensamiento y extensión. Hay, pues, una correlación entre la constitución humana y los atributos de la Sustancia que puede conocer. Sin embargo, el conocimiento humano de los atributos del pensamiento y de la extensión no implica que la sustancia se limite a tales atributos, pues su capacidad expresiva es infinita (Spinoza, 2014, E1P8S2). Los infinitos atributos de la sustancia poseen una existencia objetiva, es decir, pertenecen objetivamente a la sustancia, independientemente de sus modos¹. De forma similar, la Ontología Serial del Software no se agota en sus cuatro series actuales, pues el número de series no está predefinido. Entonces, la Ontología Serial del Software permite la adición de nuevas series sin comprometer la teoría en general.

De hecho, en el primer artículo donde se revisa, en propiedad, la Ontología Serial del Software, se pensó a modo de un preludeo para el posterior examen de la relación entre el software y el ser humano (Maldonado et al., 2020, p. 119). Es decir, primero hacía falta una revisión teórica del software desde su dimensión serial y su dimensión estructural, antes de aventurarse a examinar su relación con el ser humano en lo que equivaldría a la dimensión humana del software. Una vez concluida esta primera tarea, se procede a hacer tal examen, del cual nace la formulación de una serie adicional, a saber, la serie de la intencionalidad. El vínculo de esta cuarta serie con las demás permite una mejor comprensión del software y de su relación con el ser humano, toda vez que implica que la intencionalidad del sujeto hace parte del software, es decir, forma parte de la

¹ A menos que se haga una interpretación idealista de la teoría de los atributos de Spinoza, según la cual la existencia de los atributos no es objetiva, en la medida en que son maneras subjetivas del ser humano de percibir la sustancia. Sin embargo, para efectos prácticos, se prefiere una interpretación objetiva de los atributos.

dimensión serial y estructural del software, y no únicamente de la dimensión humana. La intencionalidad es parte integral del software.

En principio, la serie de la energía y la serie de los circuitos son de carácter físico, pues ambos son perceptibles por los sentidos, por el contrario, en tanto que no son perceptibles por los sentidos, la serie del código y la serie de la intencionalidad son de tipo simbólico. Sin embargo, de acuerdo con su carácter serial, cada serie implica a las demás, de manera que todas son, al mismo tiempo, de carácter simbólico y material. La energía, los circuitos y el código requieren de dos elementos para funcionar: información y un sustrato material. El flujo de energía circula según información que le permite ajustarse a los voltajes. No se trata, pues, de un flujo de energía azaroso, sino de un flujo en cantidad y ritmo adecuados, para lo cual se necesita de un carácter simbólico. Los circuitos, por su parte, también requieren de cierto grado de información que le permitan transmitir el código sin problemas. Gracias a este carácter simbólico, los circuitos pueden, en pocas palabras, guiar al código a través de ellos. Sin esta capacidad de transmisión, los circuitos no serían más que chatarra. Por último, la serie del código no es meramente simbólica, suposición a partir de la cual nace la famosa distinción entre hardware y software, entre lo físico y lo metafísico. Antes bien, el código posee un sustrato material en dos sentidos. Primero, en componentes físicos como la placa madre o la CPU (*programming code*), segundo, en flujos de energía organizados que viajan entre circuitos que le permiten el paso (*functioning code*). En pocas palabras, el código está hecho para una arquitectura o sustrato material específico, al cual le permite funcionar. Este funcionamiento se da gracias a que el código transmite instrucciones mientras viaja por los circuitos en forma de energía (medidas precisas de energía). Así, pues, en la medida en que convergen, estas tres series poseen una dualidad serial, esto es, simbólica y material.

Por su parte, la serie de la intencionalidad, por sí sola, parece ser solo de carácter simbólico. Pero, dado que se relaciona con las demás para dar forma al software, «la cuarta serie se materializa en su relación con las otras.» (Maldonado, 2019, p. 1), de manera que es, igualmente, de carácter simbólico y material. La intencionalidad se materializa en el flujo e interrupción de energía, es decir, en la organización de la transmisión del código, como energía, a través de los circuitos, pero también en la fabricación y en la conexión de los circuitos, y, por último, en la escritura y compilación del código en lenguaje. Así, desde la dimensión serial del software, las series son independientes ontológicamente, pero, una vez concretadas en el software, es decir, cuando el software *existe* a nivel estructural, se relacionan entre sí y dan lugar al programa en cuestión. En pocas palabras, las series son distinguibles entre sí ontológicamente, pero no están separadas ópticamente. Lo primero indica que, desde la dimensión serial, las series se pueden identificar y distinguir. Lo segundo indica que, desde la dimensión estructural, las series no están separadas en términos de existencia real o concreta. Con todo, la adición de la serie de la intencionalidad es una prueba de la flexibilidad teórica de la Ontología Serial del Software. Así, en segunda medida, la Ontología Serial del Software es flexible teóricamente en cuanto a la constitución de sus series. Queda por revisar cómo la Ontología Serial del Software se construye con el propósito de comprender la relación entre la humanidad y lo digital.

1.1.3 Comprender lo digital y sus funcionalidades

En tercer lugar, desde la propuesta de la Ontología Serial del Software se sostiene que lo digital merece una aproximación filosófica por tres motivos. Primero, porque la presencia y expansión de lo digital en la cotidianidad es evidente, en tanto que «todo se media y regula ahora

con la mencionada tecnología digital» (Maldonado, 2020, p. 17). Segundo, la filosofía, interesada siempre por la realidad como totalidad, debe acoger a lo digital como objeto de estudio en tanto que forma parte de esa realidad. Tercero, y en conexión con lo anterior, porque lo digital no solo forma parte de esa realidad, sino que, gracias a sus funcionalidades, es capaz de expandirla, y con ello, influye en nuestra forma de abordarla filosóficamente. En este sentido, «lo digital ha afectado directamente nuestra forma de pensar» (Maldonado, 2020, p. 19).

Por estos motivos, se propone una perspectiva digital desde la cual se intenta comprender ontológicamente a lo digital y que permita, asimismo, entender su relación con el ser humano. De aquí nace la propuesta de la Ontología Serial del Software (Maldonado, 2020; Maldonado & Cáceres, 2023; Maldonado & Rodríguez, 2015, 2017). Para aproximarnos a la especificidad del universo digital se requiere de su distinción ontológica con respecto al mundo analógico. Los seres humanos son seres analógicos, lo que implica un funcionamiento analógico. El universo digital se constituye como un nivel de realidad distinto (no paralelo) *en* (dentro de) el mundo analógico. Entendido así, el universo digital es trascendental e inmanente al mismo tiempo. Es trascendental en tanto que comprende elementos invisibles a la percepción humana (ceros y unos) y funciona bajo un sistema de reglas formal que le permite llevar a cabo cosas que no serían posibles en el mundo analógico. Es inmanente en la medida en que sus efectos ocurren dentro del mundo físico, lo cual es, propiamente, lo que percibimos del universo digital: «Los efectos de dichas reglas son lo único que puede percibirse del universo digital en el mundo físico: como mínimo vemos y oímos sus efectos» (Maldonado & Rodríguez, 2015, p. 33). Así, la perspectiva digital parte de una ontología del software que permite la distinción ontológica entre el universo digital y el mundo analógico. Solo después de esta ontología se pueden analizar las funcionalidades del universo digital para con los seres humanos, es decir, las maneras en las que nosotros, seres analógicos, nos

relacionamos con lo digital, y cómo, a su vez, lo digital influye en nosotros y en nuestra sociedad cada vez más digital.

Las funcionalidades del universo digital son producto de su funcionamiento según un sistema de reglas lógico-matemáticas puramente formal, las cuales están determinadas por los componentes materiales que permiten la realización física de lo digital. En palabras de los autores:

The difference between the digital universe and the analogical world allows us to understand the potentiality of the digital, that exists within the former only; it does not duplicate the world, rather it is extend it without limits under the conditions and the possibilities of the digital that are mostly logical, which, by the same token, are constrained by hardware conditions. (Maldonado & Rodríguez, 2017, p. 161).

Dado su funcionamiento a partir de un sistema simbólico de reglas formal, las funcionalidades del universo digital son impresionantes, puesto que «there are no limits for equations, for repetitions, digits or cycles» (Maldonado & Rodríguez, 2017, p. 162). Sin embargo, dichas funcionalidades están circunscritas en los componentes físicos que permiten a lo digital tener su realización física en la forma de flujo (unos) y de retención (ceros) de electricidad. Se reitera el interés por estudiar lo digital sin prescindir de sus componentes físicos, a la vez que se distingue su naturaleza ontológica con respecto al mundo analógico. Tener presente esta distinción ontológica permite dar cuenta, con mayor precisión, de las funcionalidades de lo digital. En lugar de examinar lo digital mediante los mismos parámetros de moralidad que se usan en la realidad analógica con conceptos como *bueno o malo, legal o ilegal*, entre otros, se debe examinar lo digital por sí mismo y apreciar las funcionalidades que surgen en él. Aplicarle los parámetros del mundo analógico no es más que incurrir en «una imposición de regulaciones de un mundo que ya posee unas reglas propias y de diferente naturaleza de las de un flamante universo digital» (Maldonado

& Rodríguez, 2015, p. 38). Estas categorías son aplicadas al universo digital como si le fueran inherentes, de modo tal que se ignora su naturaleza ontológica distinta del mundo analógico, así como sus funcionalidades propias.

Ahora bien, los videojuegos, en calidad de software, se insertan dentro de lo digital, de tal modo que comprender sus funcionalidades presupone, igualmente, un acercamiento ontológico. Como se señaló en la introducción, el interés académico por los videojuegos ha incrementado con el paso del tiempo. En particular, los acercamientos de corte ontológico sobre los videojuegos han tomado dos caminos: la ludología y la narratología (Cardero et al., 2014). Para la ludología, el videojuego es un sistema de reglas, de tal forma que, a partir de la comprensión de esta estructura regulada se puede comprender el videojuego. La perspectiva ludológica ve en el videojuego un artefacto que necesita ser completado (como si fuesen textos ergódicos), es decir, que funciona con base en la interactividad con el jugador. Elementos como la narrativa son secundarios al videojuego. Para la narratología, los videojuegos son, al igual que otros medios, vehículos de expresión narrativa, por lo cual es posible su comprensión ontológica a través de su narrativa. La perspectiva narratológica se fundamenta en la tesis de que los seres humanos tienen cierta inclinación cognitiva a contar historias, de manera que el videojuego se convirtió, muy pronto, en el medio ideal para la vieja actividad humana de contar historias. En todo caso, lo destacable es que los intentos por ofrecer una ontología del videojuego no son de reciente data.

Si bien los estudios sobre videojuegos han sido influenciados por las posturas ludológica y narratológica, lo cierto es que no se trata de una divergencia insalvable. Los narratólogos se han ocupado del aspecto lúdico del videojuego (Collantes, 2013; Maietti, 2013; Murray, 2017) con tanto interés como los ludólogos se han interesado por su componente narrativo (Aarseth, 1997; Frasca, 2003; Juul, 2011). En este sentido, la discusión entre la ludología y la narratología ha

cedido terreno a enfoques interdisciplinarios entre ambas corrientes y a acercamientos que buscan superar el debate. Es en este último grupo donde se ubicaría Ian Bogost. Aunque algunos lo consideran afín a la ludología (Petrowicz, 2014; Sicart, 2011), Bogost advierte que los videojuegos son «expressive cultural artifacts» (2007, p. viii), una propuesta teórica que le apunta a lo específico del videojuego. De este modo, para Bogost, estudiar los videojuegos implica, por un lado, prestar atención a la plataforma que lo soporta (2009), por el otro, comprender su capacidad de transmitir mensajes e influir culturalmente a un nivel inigualable por otros artefactos expresivos debido a su naturaleza procedural (2008c; 2007). Los videojuegos poseen una complejidad distintiva en comparación con otros artefactos expresivos. Transmiten ideas (lo cual sería su componente narrativo) y poseen mecánicas de juego (lo cual sería su componente lúdico), pero no se reducen a ninguno de los dos. A partir de esta lectura, se sugiere que la visión de Bogost sobre los videojuegos supone una ontología del videojuego en tanto software que hace compatible su teoría con la Ontología Serial del Software, de manera que le apunta, igualmente, a una comprensión de lo digital y su relación con el ser humano.

1.2 Aspectos Conceptuales

Hasta el momento se han explorado algunos de los aspectos conceptuales más relevantes de la Ontología Serial del Software, en particular la dualidad simbólica y material de sus series y la perspectiva digital que le da soporte. No obstante, aún falta revisar un último elemento de igual importancia. Se trata de la intencionalidad como la serie de mayor complejidad conceptual, en la medida en que es la que permite explorar la relación entre el software y el ser humano.

Hay tres niveles de intencionalidad, a saber, intencionalidad abierta, intencionalidad hacia abajo e intencionalidad hacia arriba. La intencionalidad abierta, también denominada intencionalidad general, se refiere a la propiedad de la computación digital de no tener un propósito o función específica inherentemente definida, sino más bien la capacidad de adaptarse y de realizar una amplia gama de tareas o funciones según la programación o las instrucciones proporcionadas. Desde la Ontología Serial del Software se propone entender la intencionalidad general a partir de la metáfora de la máquina universal de Turing o MUT (Maldonado, 2019). Esta máquina es capaz de llevar a cabo una tarea R, mediante la simulación del diagrama M de una máquina de Turing particular MT. Así, la MUT tendrá su propio diagrama N, que no es más que, en este caso, la simulación del diagrama M, en la medida en que se propone llevar a cabo la tarea R para la cual fue diseñada MT. La operación quedaría formalizada de la siguiente manera: $MUT = N (M(R))$. Así, la MUT es capaz de simular, en su diagrama N, el diagrama M de la MT, pero no solo ese, sino que, dado cualquier tarea que se le presente, sea R o cualquier otra, podrá resolverla mediante la simulación del diagrama de una MT particular capaz de resolver esa tarea particular. De ahí que, según esta lectura, lo más cercano a una máquina universal de Turing sea el computador digital, en tanto que posee la capacidad de ejecutar múltiples programas. Cada programa, esto es, cada software en particular, sería el equivalente de una máquina de Turing. Un computador digital es una máquina universal de Turing que los usuarios podrían programar a su acomodo. La intencionalidad abierta del computador digital le permite albergar programas particulares, siempre y cuando estos se mantengan, por decirlo de alguna manera, dentro de los parámetros de la intencionalidad abierta del computador. Y es que la intencionalidad de ser programado fue previamente inscrita en el computador a nivel de la serie de los circuitos, de la energía y del código,

y a la cual el usuario-programador tendrá que ajustarse al momento de escribir código de alto nivel, es decir, al momento de crear programas particulares para el computador digital.

La intencionalidad hacia abajo da cuenta del fenómeno humano de la sobre-codificación, ahora trasladado a lo digital (Maldonado, 2019). Basado en la teoría de Searle sobre los actos de habla, en particular su fórmula «X cuenta como Y en el contexto C» (Searle, 2017), la sobre-codificación es el acto de darle un significado nuevo a algo que no lo tenía, de manera que lo reemplaza. Este reemplazo se puede dar de manera indefinida. Si X se asume como si fuese Y en el contexto C, entonces Y tomará el lugar de X. Después, esta Y será la X en un nuevo contexto, en donde, además, se tomará como una nueva Y, y así sucesivamente. La sobre-codificación invisibiliza el contexto original de X, lo que arroja luz sobre dos fenómenos: explica, primero, por qué motivo la intencionalidad en el software suele pasar desapercibida; segundo, las razones por las que el software se suele entender como mero código. La explicación de ambos fenómenos proviene de una misma raíz, a saber, el computador digital posee, fundamentalmente, una intencionalidad abierta, la cual es sobre-codificada, y de donde aparecen los cinco niveles distintos de programación del código (*programming code*), a saber, el código de bajo nivel, el código de la CPU, el código del sistema operativo (OS), el código del compilador y el código de alto nivel. Este último es el código con el que el usuario-programador, propiamente, interactúa. Cada nivel invisibiliza al anterior, de manera que el usuario solo percibe la superficie. Por otro lado, en tanto que en los niveles más bajos del código se parte de la codificación de lo que se ha denominado sustrato material (CPU, placa madre, circuitos, entre otros, pues el código de los niveles más bajos fue, previamente, inscrito en estos componentes físicos), y estos niveles del código son invisibilizados por los de más alto nivel, se ignora que en el computador digital hay código e intencionalidad en lo físico, y sin lo cual no le es posible funcionar. Todo el computador digital

está codificado, lo que implica múltiples intencionalidades. De este modo, se invisibiliza el hecho de que, antes del código de alto nivel, es decir, antes de que el usuario-programador pueda programar, hay otras capas de intencionalidades subyacentes. Los lenguajes de programación, por ejemplo, fueron previamente programados, pero la sobre-codificación invisibiliza este hecho. Se asume que todo el sustrato material, los lenguajes de programación y demás ya están dados de antemano sin ninguna intencionalidad. La Ontología Serial del Software advierte que la intencionalidad no es singular, sino que existe en capas de intencionalidad superpuestas entre sí. En efecto, el usuario-programador de un lenguaje de programación utiliza un código de alto nivel previamente programado, es decir, con una intencionalidad previa, pero también utiliza un sistema operativo, el cual fue igualmente programado con anterioridad para ser capaz de ejecutar ese lenguaje de programación, y así sucesivamente.

La intencionalidad hacia arriba indica que el propósito específico de un software particular, previamente insertado por sus programadores, puede cambiar a nivel de usuario. Después de todo, es el usuario quien, en últimas, le da un propósito al software mediante el uso. Así, un software no es independiente del usuario, pues es el usuario quien implica la existencia del software (Maldonado, 2019). Pero estos propósitos del usuario no se escapan al propósito específico del programa, sino que se circunscriben dentro de sus parámetros, de manera similar a como los programas particulares se circunscriben al computador digital. Esto se debe a que la intencionalidad general del computador digital está recortada, es decir, si bien tiene la capacidad para realizar una amplia gama de tareas, esta capacidad se enfoca en funciones específicas a través de la programación de software, lo que da como resultado programas individuales con intencionalidades más concretas y específicas. Los programas particulares son, pues, recortes de la intencionalidad general. En efecto, antes de que el programa sea utilizado por el usuario en un

computador digital, ya hay una serie de capas de intencionalidad que permiten ese momento. No obstante, esto no implica que la intencionalidad abierta, ahora recortada, sea una intencionalidad cerrada. Al contrario, todavía es una intencionalidad abierta en la medida en que el usuario concretiza el propósito específico del software particular. Como ejemplo, se revisa el caso de los programas de lectura de archivos de texto digitales (Maldonado, 2019). Los archivos de texto, cada uno de los cuales cuenta con una extensión particular (pdf, doc, docx, epub, etc.), solo son fragmentos del programa que es capaz de reproducirlos. El propósito específico del programa, esto es, leer archivos de texto digitales, se completa con la acción del usuario. Lo que ocurre al abrir un archivo de texto digital es que el usuario concretiza el propósito específico del programa al otorgarle un uso, lo que completa el programa con un fragmento de sí mismo, a saber, el archivo de texto digital.

Es en el tercer nivel de la intencionalidad donde se encuentra al usuario, quien puede tomar, o bien la subjetividad de usuario-consumidor, o bien la de usuario-programador, o bien una combinación de las dos. El usuario programador es quien despliega, mediante la modificación del código, el propósito del software, aunque lo hace de acuerdo con las capas de intencionalidad que le preceden, como se vio en el segundo nivel de la intencionalidad. El usuario programador no puede, por sí solo, escribir todo el código de un software; en vez de eso, programa mediante el uso de (de ahí que se le denomine usuario) lo producido por programadores previos en un computador digital previamente dispuesto a ello. Esto es lo que se debe entender por programar mediante el uso de un código de alto nivel. Por su parte, el usuario consumidor es quien concretiza el propósito específico del software, esto es, quien le otorga al software un uso, completándolo en el proceso.

Ahora bien, si la intencionalidad humana hace parte integral del software, es decir, se encuentra, junto a las otras tres series, en las dimensiones serial y estructural del software, ¿qué

ocurre con la llamada dimensión humana? Como se señaló en su momento, la intencionalidad fue pensada, en principio, como parte de la dimensión humana, pero, tras varios análisis, se concluyó que también forma parte del software a nivel serial y estructural como una serie más. Esto quiere decir que la intencionalidad se despliega en las tres dimensiones, es decir, forma parte tanto del software en sí mismo como de la dimensión humana. De hecho, desde la propuesta de la Ontología Serial del Software se ofreció, en su momento, una clasificación tentativa de usuarios desde la dimensión humana (Maldonado & Rodríguez, 2015). Esta clasificación preliminar sugiere los siguientes niveles de usuario: *click and play*, *plug and play*, *follow instructions and play*, *run and play*, *program and play* y *hack and play*. Los primeros niveles comprenden el uso básico de los programas, mientras que los demás, en particular los dos últimos, demandan mayor perspicacia y dominio en los usuarios, en la medida en que devienen programadores. En el último nivel se ubica a los programadores de sistemas operativos y de lenguajes de programación, de modo que quien esté en este nivel es más un programador que un usuario, si bien todavía se le considera un usuario, dada la tesis de las capas de la intencionalidad discutida previamente. Esta clasificación amerita una actualización, pues cuando se formuló aún no se tenía clara la presencia de la intencionalidad a nivel de las series y estructura del software. Lo cierto es que el despliegue de la intencionalidad en las tres dimensiones advierte que la Ontología Serial del Software, esto es, el software en las dimensiones serial y estructural, está enmarcada dentro de una ontología social, en consonancia con la propuesta de Searle sobre la construcción de la realidad social (1997, 2017). En efecto, la realidad social se construye a partir de, entre otros, la constante sobre-codificación humana, fenómeno que se ha trasladado a lo digital. Con todo, revisar esta parte de la teoría de la Ontología Serial del Software está fuera del alcance de los objetivos de este trabajo, más aún cuando no se cuenta con material de referencia que la explore al detalle. En última instancia, dado que la

Ontología Serial del Software todavía se encuentra en desarrollo, solo se espera hacer una modesta contribución mediante su uso como método para el análisis de una parte de la teoría de un autor particular cuya producción literaria, igualmente, continúa. La labor investigativa no se agota con lo expuesto aquí, por el contrario, mantiene su apertura a futuras investigaciones complementarias.

Con todo, se espera llevar a cabo una revisión satisfactoria de la teoría de Bogost por medio de todo lo expuesto a lo largo de este capítulo. Se destacan dos aspectos clave como herramientas valiosas para la tarea. Primero, la ausencia de una distinción entre hardware y software, lo cual sugiere que el estudio de un software comprende la revisión de sus elementos simbólicos y materiales, sin una distinción especial entre ellos. Segundo, la importancia de la serie de la intencionalidad en la comprensión del software, su funcionamiento y su posición dentro de la relación entre la humanidad y el universo digital. Se intentará dar cuenta de qué tan cercana es la teoría de Bogost a estas formulaciones, y qué tan útil resulta para la comprensión del fenómeno del videojuego en esta sociedad digital. Por lo pronto, se procede a exponer y estructurar la teoría de Bogost en el segundo capítulo.

2. Una Aproximación a la Propuesta Teórica de Bogost

Ian Bogost posee, principalmente, una formación académica en humanidades. Graduado en Filosofía y Literatura Comparativa en la Universidad del Sur de California USC, siguió por el mismo camino hasta obtener un doctorado en Literatura Comparativa en la Universidad de California en los Ángeles UCLA. Desde entonces, Bogost se dedica a la docencia universitaria en literatura, pero también al estudio y desarrollo de videojuegos, hecho por el cual, hoy en día, es más reconocido (Williams, 2019). A su formación en humanidades se une su particular interés por la propuesta teórica de Graham Harman, la *ontología orientada a objetos (OOO)*, a la cual recurre en varias ocasiones. Entonces, la literatura comparativa y la *ontología orientada a objetos* se identifican como las mayores influencias en la obra de Bogost.

Ahora bien, la teoría de Bogost no es uniforme, pues se ha ocupado de multitud de temas. Para la delimitación de las obras a estudiar se tiene en cuenta el interés particular de este estudio, a saber, apuntar hacia una ontología del videojuego, de modo que se acude a las obras que sugieran respuestas a la pregunta ¿qué es un videojuego? En este sentido, se procura brindar cohesión a esa parte de la teoría de Bogost que será analizada, y así facilitar su posterior estudio a través del enfoque de la Ontología Serial del Software.

2.1 Primer Método de Análisis: Estudios de Plataformas

Una de las primeras incursiones de Bogost en el estudio de los videojuegos se produjo con la publicación de *Racing the Beam* para *La Serie de Libros de Estudios sobre Plataformas (The*

Platform Studies Book Series). El libro es un estudio de la Atari VCS como plataforma, es decir, como un conjunto de componentes de hardware, de software y modulares (periféricos) que definen las posibilidades del proceso creativo y, asimismo, la recepción de su resultado. Bogost denomina a este modelo de análisis *estudios de plataformas (platform studies)*, a la cual describe de la siguiente manera: «the perspective of platforms as one way, one productive way to think about the coherent sets of shared constraints that tend to be packaged in hardware or software form» (CU Boulder Libraries, 2011, 36:55-37:06). Los estudios de plataformas son, pues, una apuesta por entender la manera en que el hardware y el software de una plataforma influyen en el proceso creativo. En este sentido, la Atari VCS es solo una plataforma en concreto susceptible de este tipo de análisis, aunque su elección como objeto de estudio es premeditada, como se verá más adelante.

Por otro lado, el estudio de una plataforma implica, a su vez, una revisión del contexto cultural que la vio nacer. Los componentes disponibles para su manufactura, la complejidad y características de su código, el diseño de la interfaz y de las formas utilizadas, así como el uso final que se le da a la plataforma son todos elementos fundamentales para su comprensión. En palabras de Bogost, los estudios de plataformas conectan «the fundamentals of digital media work to the cultures in which that work was done and in which coding, forms, interfaces, and eventual use are layered upon them.» (Montfort & Bogost, 2009, p. 147). En lo que sigue se propone una revisión de los puntos centrales de esta propuesta.

2.1.1 Atari VCS: Trazos sobre Qué Es un Videojuego

La elección de la Atari VCS como la plataforma objeto de estudio le permite a Bogost formular una interpretación de la historia de los videojuegos donde tecnología y cultura se

entrelazan. De esta interpretación se pueden extraer ciertas ideas sobre qué es un videojuego. Los videojuegos de la Atari VCS no son, en su mayoría, originales, sino que se trata de versiones de videojuegos arcade adaptados a la consola, lo que se conoce como *ports*. La consola es, de hecho, una apuesta comercial por expandir el alcance de los videojuegos de los bares y zonas recreativas (arcades) al ámbito casero. Así, Bogost (2009, pp. 6-11) es capaz de construir una sólida interpretación de la historia del desarrollo de videojuegos: en principio, los videojuegos surgieron de la experimentación con la tecnología disponible, como es el caso de *Spacewar* en el minicomputador PDP-1 y un osciloscopio, que fue exhibido entre los estudiantes en el MIT, hasta que, poco a poco, se abrieron camino a la comercialización en masa. En este sentido, el desarrollo de videojuegos estuvo directamente influenciado por la tecnología disponible.

Ahora bien, de acuerdo con Bogost (Montfort & Bogost, 2009), los videojuegos arcade fueron diseñados con lo que en psicología se conoce como *refuerzo parcial* (*partial reinforcement*), un efecto psicológico que busca generar apego y, en muchos casos, adicción a una actividad. Esta estrategia es derivada de los juegos de tragaperras y juegos de feria (*midway games*), donde se emplean elementos que estimulan el interés y la continua participación. Los videojuegos arcade adoptaron esta característica, no solo para fomentar el consumo del juego en sí, sino también para promover el consumo de productos y servicios propios del lugar, lo que generó una especie de vínculo simbiótico entre el videojuego y su entorno. Más tarde se lanzó la Magnavox Odyssey, la primera consola de videojuegos casera, seguida muy de cerca por la Atari VCS. Las dos consolas adaptaron a su manera los videojuegos arcade más populares, lo que sugiere una continuidad en el desarrollo de videojuegos, impulsada por intereses económicos. En efecto, la selección de los videojuegos arcade para trasladar a la plataforma fue deliberada, y la idea de incluir a familias en el consumo de videojuegos y no estancarse entre los visitantes recurrentes de arcades, dan cuenta

del interés económico detrás y de un cambio en la manera como se experimentaban los videojuegos. Se pasó de la cultura propia de bares y tabernas al entorno doméstico.

Tanto la Magnavox Odyssey como la Atari VCS son el intento por expandir el alcance de los videojuegos, esta vez, al ámbito doméstico. ¿Y qué mejor manera de lograr este objetivo que por medio del uso de un aparato que, por un lado, es capaz de suplir la necesidad de generar una imagen para el videojuego y, por el otro, era ya bastante común entre las familias norteamericanas: la televisión? De hecho, Bogost sugiere que es a partir de esta comunión que se popularizó el término videojuego: «The focus on the production of images for display on the TV helps explain why games running on circuits and later computers became known as ‘videogames’» (2009, p. 14). Resulta llamativa esta interpretación del surgimiento y acentuación del término videojuego, en particular porque parece ser un antecedente a propuestas narratológicas como la de Collantes. Para Collantes (2013), los juegos y los relatos pertenecen a una misma categoría a nivel narrativo, pero se distinguen porque en los primeros predomina la narración performativa (el individuo participa de la narración mediante la apropiación y el desempeño de un rol), mientras que en los segundos predomina la narración enunciativa (posee características enunciativas que regulan su aproximación). Collantes sugiere que los videojuegos, en tanto que se desenvuelven en una pantalla bajo un punto de vista predefinido por un autor (desarrolladores), el videojuego siempre tiene, al menos, ese componente enunciativo, propio de la enunciación audiovisual. Según esta perspectiva, el videojuego se distingue de los juegos en la medida en que predomina la narración performativa, pero siempre le es propio, al menos, este elemento de tipo narración enunciativa.

Entonces, por el momento, se tienen las siguientes características del videojuego: (i) desde su aparición, el videojuego estuvo directamente relacionado con el desarrollo tecnológico del momento (de osciloscopios a la televisión) y con el contexto cultural (de arcades al entorno

doméstico), (ii) existe una continuidad en el desarrollo de videojuegos que suele ser impulsada por motivaciones económicas a las que debe, en buena medida, su popularidad, y (iii) el término *videojuego* está estrechamente relacionado con la generación de imagen, en particular la generada a través del uso de la televisión.

2.1.2 El Proceso Creativo en la Atari VCS

La Atari VCS es una consola compuesta por componentes cuidadosamente seleccionados. Todos estos componentes son de bajo costo, precisamente para que el precio final de la consola fuera asequible y, con ello, lograr conquistar el entorno doméstico. Pero, además, funcionan de una manera muy cohesiva en su conjunto, es decir, cada componente cumple con su debida función y determinan, en su totalidad, lo que la consola puede y no puede hacer. Esto representa, de entrada, el mayor obstáculo de los desarrolladores de videojuegos que deseen publicar un título en la consola: deben adaptarse a sus limitaciones técnicas. Un ejemplo de estas limitaciones es la representación de imágenes. La cantidad de RAM de la Atari VCS era de tan solo 128 bytes, de modo que no tenía suficiente espacio para almacenar los datos de todo un fotograma en la pantalla. En su lugar, las imágenes son dibujadas línea por línea en un proceso que ajusta la velocidad de dibujo del electrón de la TV (propio de la tecnología CRT de la época), de manera que se acelere o desacelere, según sea necesario. Así, pues, los desarrolladores debían entender este proceso de generación de imagen, y ajustar su código al movimiento del electrón:

This task requires that the programmer write carefully timed code that fits the motion of the television's electron beam. Some graphical effects demand changes to the TIA's registers in the

middle of a single scan line. In these cases, the programmer must carefully “cycle count” processor instructions so they execute at the right time. (Montfort & Bogost, 2009, p. 28)

Ahora bien, estas restricciones no deben interpretarse de manera peyorativa. Es cierto que los desarrolladores deben ajustarse a ellas, pero, al mismo tiempo, son aquello que permite el proceso creativo en primer lugar. El resultado de este proceso, el videojuego, revela las capacidades de la plataforma que lo soporta. En palabras de Bogost: «The work that's built for a platform is sort of supported and constrained by what that platform can and can't do» (CU Boulder Libraries, 2011, 38:37-38:45). En este sentido, resulta más apropiado pensar en los componentes de la plataforma como el elemento fundamental que da forma al proceso creativo, y no como su mayor obstáculo.

Las limitaciones técnicas de la Atari VCS desafiaron a los desarrolladores a encontrar soluciones innovadoras, a explorar nuevas formas de expresión, de acuerdo con lo que deseaban plasmar en su videojuego, y a utilizar de manera óptima los recursos disponibles. Esto provocó, a su vez, que los videojuegos de la plataforma tuvieran sus propias características que los diferenciaban de sus versiones arcade. Pero no se trata solo de características superficiales, como la apariencia, fácilmente detectable por el jugador que haya probado de primera mano el arcade del que se deriva el videojuego de la Atari VCS en cuestión, sino de características del código, que son más profundas, como la manera en que funcionan los elementos del juego, es decir, cómo se mueven y cómo interactúan con el jugador, lo que provoca, a su vez, experiencias significativamente diferentes en los jugadores. Un ejemplo representativo de esto es el de la versión de la Atari VCS de *Asteroids*, donde el comportamiento de los asteroides es distinto tanto al nivel de las direcciones en las que se mueven, arriba y abajo, más un ligero movimiento horizontal, en contraste con el movimiento arbitrario y a distinta velocidad de los de su contraparte

arcade, lo que viabilizó la estrategia de permanecer en el medio sin necesidad de utilizar la mecánica del empujón (Montfort & Bogost, 2009). El resultado de estas variaciones, esto es, la manera como los jugadores experimentan los videojuegos, da cuenta de una característica adicional de los videojuegos a la que Bogost tratará con mayor propiedad en sus obras posteriores: la expresividad, entendida, por el momento, como la influencia de los videojuegos en sus jugadores. Para ello ha de revisarse, primero, otro de sus modelos de análisis, a saber, el de la operación unitaria (*unit operation*).

2.2 Segundo método de análisis: la proceduralidad

La formación en literatura comparativa vuelve a cobrar protagonismo en el análisis de Bogost a los videojuegos, en este caso, con la formulación del concepto de operación unitaria, el cual le servirá para apostar en favor de una crítica comparativa (*comparative criticism*) de múltiples artefactos de expresión cultural: el cine, la literatura, el arte y los videojuegos. Al mismo tiempo, le permite proponer una nueva forma de analizar cada artefacto de manera individual, denominada análisis unitario (*unit analysis*). Los alcances de esta propuesta son, como es evidente, bastante amplios, muy cerca de escaparse del marco de estudio de los videojuegos, pero resultan esenciales para comprender la interpretación de Bogost de los videojuegos como artefactos de expresión cultural y, más adelante, artefactos persuasivos. A continuación, se revisa la tríada medular de esta propuesta, a saber, operación unitaria, análisis unitario y crítica comparativa, para luego revisar detenidamente la propuesta de los juegos persuasivos.

2.2.1 Operación Unitaria

Como concepto, la operación unitaria se puede entender como la combinación de dos elementos fundamentales. Por un lado, es unitaria, lo que implica que es una unidad o entidad individual, y, por el otro, es una operación, que se refiere a un proceso que sigue un conjunto de reglas internas en lo que se denomina procedimiento (*procedure*). Inspirado por la ontología orientada a objetos (*object oriented ontology OOO*) y por la teoría de los sistemas complejos (*complex system theory*), Bogost define la operación unitaria como «a general conceptual frame for discrete, compressed elements of fungible meaning» (Bogost, 2008c, p. xiii). Una operación unitaria tiene tres características: (i) la discreción, lo que significa que se puede distinguir de otras operaciones unitarias en tanto que es única, (ii) la compresión, lo que implica que tiene un significado comprimido en sí misma, de tal forma que se puede entender y analizar de forma autónoma a partir de su propia lógica interna, y, finalmente, (iii) la conmutabilidad, o lo que es lo mismo, la capacidad de formar parte de un sistema y de transferirse entre sistemas, por lo cual es posible que sean «encapsulated and transmitted across media» (Bogost, 2008a, p. 11). Entonces, La operación unitaria es discreta y conmutable, es decir, no está sujeta a un sistema complejo, sino que puede ser transmitida a otros sistemas sin perder su unicidad. De esta manera, una operación unitaria es conmutable como un objeto de la *OOO* es polimórfico. Además, la operación unitaria también implica la compresión, de modo que ella misma puede albergar a otras operaciones unitarias como un sistema complejo, sin perder su calidad de operación unitaria.

Cabe detenerse en lo que Bogost entiende por sistema. Aunque Bogost nunca ofrece una definición literal del término, sí es posible hacerse una idea a partir de los ejemplos que ofrece, como el funcionamiento de un motor, las políticas de una empresa o la estructura y normativa de

un ejército (2008b). Considera sistemas mecánicos, organizacionales o conceptuales, de modo que cualquier actividad social, política o cultural puede tratarse como un sistema complejo, esto es, como un set de operaciones unitarias. Esto vuelve susceptibles a estas actividades a una representación procedural y, por extensión, a una representación particular en la que se sugiera cómo funciona o cómo debería funcionar dicho sistema, por medio de la retórica procedural. Casos concretos de retórica procedural en videojuegos los encontramos en *The McDonald's Videogame*, el cual representa el funcionamiento de una empresa y elabora aseveraciones sobre sus prácticas cuestionables, o en *America's Army*, el cual representa las normas y la estructura jerárquica dentro del ejército americano, e incita a la reflexión sobre el funcionamiento ideológico de ese sistema en la medida en que cualquier incumplimiento de las normas se puede considerar como un acto de insubordinación.

Esta propuesta conceptual es retomada por Bogost en *Alien Phenomenology* (2012) con consecuencias filosóficas interesantes, sin embargo, allí su enfoque es más amplio en tanto que se propone construir, siguiendo a Harman, una ontología orientada a objetos, es decir, una ontología no humana. Por el momento, se destaca la flexibilidad del concepto y de las aplicaciones que Bogost le otorga, en particular para el análisis de videojuegos.

2.2.2 Análisis Unitario

Dado que los artefactos expresivos pueden estar configurados por operaciones unitarias, la labor de un crítico de estos artefactos será la de revelar tales operaciones unitarias (Bogost, 2008c). El análisis unitario es un método de análisis que consiste en extraer las operaciones unitarias de un artefacto y analizarlo en consecuencia, puesto que «unit operations give us a lever for

understanding any form of human production as potentially procedural.» (Bogost, 2008c, p. 15). En contraste con el enfoque de las operaciones del sistema (*system operations*), la operación unitaria permite abordar las unidades individuales de un sistema. Esto implica una mirada detallada a los elementos específicos que componen un artefacto, en lugar de un enfoque holístico del sistema en su totalidad.

Aplicar el análisis unitario a un artefacto no implica invalidar cualquier otro método de análisis, se trata, más bien, de tomar el artefacto como un sistema complejo con operaciones unitarias. Sin embargo, en algunas ocasiones, los resultados del análisis unitario de un artefacto resultan más significativos que los de otros tipos de análisis. Un ejemplo de lo anterior se evidencia en *The Terminal*, película a la que Bogost (2008c, pp. 15-19) aplica un análisis unitario para revelar que se trata, fundamentalmente, de varios modos de espera no corroborada que se entrelazan entre sí. Este enfoque difiere de otras interpretaciones: por un lado, la interpretación de una operación del sistema considera a la película como la lucha de un hombre contra un gobierno por su identidad, de modo que intenta darle un sentido narrativo a nivel general, por el otro, la interpretación narrativa individual que enfatiza en las historias de cada personaje y las conecta con la espera no corroborada, relegando a esta última como secundaria.

Entendida como una película sobre la operación unitaria de la espera no corroborada en múltiples formas, es decir, como múltiples operaciones unitarias (la espera no corroborada en cada personaje particular) que forman parte, a su vez, de una operación unitaria (la espera no corroborada en general), *The Terminal* adquiere un carácter expresivo en la medida en que invita a la reflexión sobre la espera no corroborada desde la experiencia personal del espectador. El propio Bogost, al analizar la película, sintetiza su experiencia de la siguiente manera: «I watched it a second time on a transatlantic flight. The function of the in-flight movie itself is a medium for

waiting; it is provided to distract passengers as they wait for the next milestone in the flight.» (2008c, p. 18). Esta misma expresividad es propia de los videojuegos, como se verá más adelante.

2.2.3 Crítica Comparativa

La crítica comparativa es una metodología que permite analizar varios artefactos de expresión cultural al mismo tiempo a partir de un elemento en común, a saber, las operaciones unitarias. La crítica comparativa es la aplicación del análisis unitario a varios artefactos de expresión cultural, de tal forma que se propone comprender cómo estas operaciones unitarias han sido abordadas a lo largo de los años y desde diferentes formas de expresión. Bogost (2008c, pp. 73-89) emplea esta metodología en su estudio comparativo de cuatro artefactos, que son: *A une passante*, un poema de Baudelaire, *A woman on the street*, un poema de Bukowski, *Amélie*, una película de Jeanet, y el videojuego *Los Sims*, particularmente su paquete de expansión *Hot Date*.

Para los propósitos de este trabajo, lo más relevante de esta aplicación de la crítica comparativa es que se constituye como un ejemplo de la conmutabilidad de las operaciones unitarias dada su recurrencia en distintos artefactos. En este caso, la operación unitaria son los encuentros fortuitos. Este elemento en común es desarrollado por cada artefacto de una manera muy particular, esto es, mediante el lenguaje escrito en los dos primeros, mediante la narrativa audiovisual en la película y mediante la interacción en *Los Sims*. El carácter interactivo del videojuego les permite a los jugadores tomar decisiones sobre cómo llevar a cabo el encuentro fortuito, lo que fomenta la reflexividad sobre este sistema del mundo real, previamente plasmado por Baudelaire en un poema, ahora trasladado al entorno digital: «By laying bare the figure of the chance encounter in the form of software, *The Sims* invites players to examine their own

satisfaction with this 150-year-old social rubric» (Bogost, 2008c, p. 89). Queda por discutir si esta metodología es útil para el análisis de artefactos de expresión cultural en general, o si, más bien, solo es particularmente valiosa para el campo de los videojuegos, como sugiere Corona (2014). Lo cierto es que la interactividad es, de hecho, una de las principales características del videojuego que da lugar a su expresividad, es decir, a su incidencia en la experiencia humana.

2.2.4 Juegos Persuasivos

La propuesta de los juegos persuasivos surge como respuesta a dos aspectos clave en la vida de Bogost: por un lado, su creciente interés por estudiar un medio donde se encuentren la computación y la cultura, y por el otro, su marcada decepción y escepticismo hacia la cada vez más popular propuesta de los juegos serios. El interés por los videojuegos como medio computacional y cultural fue explicitado por Bogost en una entrevista con Williams (2019): «I was interested in computing and I was interested in cultural studies, and videogames presented themselves as one of the only candidates where that marriage was taking place» (p. 470). Esta unión entre computación y cultura en los videojuegos se ve reflejada desde su creación en el código, pues incorporan elementos culturales de la mano de sus desarrolladores y, cuando son operados, es decir, cuando se juegan, tienen un impacto cultural en sus jugadores. En este sentido, los videojuegos son artefactos de expresión cultural capaces de representar fenómenos del mundo real en un entorno digital, de manera que influyen en la perspectiva de los jugadores sobre dichos fenómenos.

En segundo lugar, los juegos persuasivos aparecen como una respuesta al creciente interés por los juegos serios. Los juegos serios es un movimiento inaugurado por Clark Abt en su libro

Serious Games (1987), el cual consiste en otorgar a los juegos una finalidad importante, es decir, *seria*, que les permita cobrar relevancia más allá de su carácter divertido. Para Bogost (2007) la propuesta de los juegos serios, la cual empezó a cobrar relevancia a principios del siglo XXI, constituye un intento fallido por abordar con propiedad a los videojuegos, pues perpetúa el dualismo entre lo serio (generalmente considerado aburrido) y lo no-serio (generalmente asociado con la diversión). Como resultado de esta dicotomía, los juegos serios sirven de soporte para los intereses particulares de las instituciones que los financian. Esta dicotomía centra su atención en los intereses particulares de las instituciones y en separarlos de elementos como el entretenimiento, en lugar de centrarse en la capacidad expresiva de los videojuegos.

Una aproximación más adecuada a los videojuegos es aquella que dé cuenta de su expresividad, es decir, de su capacidad de influir en los jugadores mediante la transmisión de ideas, emociones y conceptos, de tal forma que se supere la dicotomía entre lo serio y lo no serio. Los videojuegos ya son inherentemente serios. De aquí nace la propuesta teórica de Bogost: los juegos persuasivos, un tipo de videojuego que lleva a cabo un intento de persuasión por medio de la retórica procedural.

2.2.4.1 Juegos Persuasivos: Presupuestos Conceptuales. Como categoría de videojuegos, los juegos persuasivos son videojuegos que aplican con eficacia la retórica procedural (*procedural rhetoric*), un tipo de retórica propuesta por Bogost y fundamentada, por un lado, en el estudio de Janet Murray (2017) sobre la proceduralidad (*procedurality*) propia de la computación y, por el otro, en estudios recientes sobre retórica, en particular los de Kenneth Burke (1969). En este sentido, la retórica procedural es la combinación de dos elementos: por un lado, de una comprensión de las características del medio en particular, en este caso, los videojuegos y su

carácter fundamentalmente procedural, y, por el otro, de la práctica general de la retórica, que implica persuasión y expresión (Bogost & Wright, 2007). A continuación, se revisarán estos dos elementos con más detalle.

2.2.4.1.1 Proceduralidad. Un proceso es la ejecución de una actividad de acuerdo con una serie de reglas, esto es, a una lógica interna que determina la manera como se lleva a cabo la actividad en cuestión. Bogost (2007) entiende al proceso como una operación unitaria, lo que le confiere las características de la discreción, la compresión y la conmutabilidad. Ahora bien, la proceduralidad es la capacidad de autoría de estos procesos, de manera que la proceduralidad es propia de los medios que funcionan, fundamentalmente, por medio de procesos, es decir, funcionan de manera procedural. Lo anterior sirve como fundamento del modelo de análisis de la operación unitaria, es decir, en la lectura de artefactos expresivos como potencialmente procedurales en la medida en que, tras la aplicación de un análisis unitario, se revelan sus operaciones unitarias, o lo que es lo mismo, se revela su funcionamiento regulado. Sin embargo, aunque la proceduralidad pueda encontrarse tanto en medios computacionales como no computacionales (pues, de otro modo, la propuesta del análisis unitario y de la crítica comparativa no tendrían razón de ser), es solo en la computación donde la representación procedural (*procedural representation*) tiene lugar. En palabras de Bogost (2007), la representación procedural «explains processes with other processes. Procedural representation is a form of symbolic expression that uses processes rather than language.» (p. 9). En efecto, los medios computacionales no solo funcionan proceduralmente, sino que, además, son capaces de representar, también proceduralmente, el funcionamiento procedural de otros medios no computacionales, o lo que es lo mismo, usan procesos para representar otros procesos.

Cuando se afirma que la proceduralidad de la computación representa procesos de un sistema, se sostiene que se hace una transferencia de esos procesos a código, en lugar de simplificarlos. Se crea una representación digital del proceso original, adaptada al entorno computacional. En la transferencia se pueden realizar cambios en la representación digital para adecuarla al software y a los objetivos específicos que se persiguen. En palabras de Bogost: «The methods that compose software objects thus do not necessarily condense the complexities of the natural world via mathematical or symbolic reductions; rather, they encapsulate representations of the world into specific software structures» (2007, p. 40). Una vez en el entorno digital, es posible interactuar con los procesos de una manera similar a como se haría con el proceso original. En el caso de los videojuegos, esta exploración se lleva a cabo en el acto de jugar. De esta manera, la computación proporciona un entorno controlado y predecible para estudiar y representar procesos, pues mediante el código de los programas informáticos se pueden ejecutar diversos procesos de forma sistemática y reproducible. Esta capacidad de representar proceduralmente permite a los medios computacionales, entre otras cosas, crear simulaciones concretas sobre otros sistemas, si bien esto se hace bajo ciertas modificaciones, pues, de acuerdo con Bogost (2007), una simulación nunca será, ni pretende serlo, una imitación completamente fiel del original.

Si los medios computacionales son fundamentalmente procedurales y les es propio la representación procedural, cabe preguntarse por qué Bogost muestra un interés particular por los videojuegos dentro de la amplia gama de programas computacionales. La respuesta (2007, pp. 44-45) se puede sintetizar en cuatro aspectos clave. Primero, si bien funcionan en código como los demás artefactos computacionales, los videojuegos tienden a tener un código más complejo, lo que implica mayor demanda de capacidad computacional. Segundo, los videojuegos tienen objetivos similares a los de otros medios no computacionales, es decir, pueden evocar o transmitir ideas,

emociones o conceptos. En otras palabras, los videojuegos son expresivos. Tercero, los videojuegos requieren de un alto grado de interactividad (*interactivity*) en el acto de jugar a partir del cual, por ejemplo, los juegos persuasivos logran completar su retórica procedural: «video games more frequently and more deeply exploit the property of the computer that creates the kind of possibility spaces that we can explore through play.» (Bogost, 2008b, p. 122). Se revisará esta característica con mayor profundidad más adelante. Por último, Bogost elige enfocarse en los videojuegos debido a su propia preferencia personal, ya que él mismo es jugador, crítico y desarrollador de videojuegos.

Por otro lado, y en diálogo con su formación académica en literatura, Bogost (2007) propone que, así como hay figuras literarias como la metáfora o la metonimia, las cuales constituyen toda producción literaria, también existen figuras procedurales, como oprimir un botón o la barra de desplazamiento. En los videojuegos, el movimiento o la detección de colisiones se pueden tomar como ejemplos de figuras procedurales. Estas figuras sirven, en ambos casos, para crear operaciones unitarias.

Se propone revisar el caso particular de los videojuegos. Las figuras procedurales como el salto o la detección de colisiones permiten la autoría de operaciones unitarias específicas en el videojuego. Por ejemplo, de la aplicación de la figura procedural del salto en un videojuego de plataformas surge la operación unitaria específica de superar obstáculos mediante el salto. La operación unitaria se convierte en una parte fundamental de la mecánica de juego al definir cómo el personaje reacciona a los obstáculos. Cuando el jugador presiona el botón de salto, el personaje ejecuta esta acción y salta sobre el obstáculo, esto es, la figura procedural del salto pasa a convertirse en una operación unitaria específica. En este sentido, en el caso de los videojuegos, como bien señala Petrowicz, es posible entender el término "operaciones" de manera más literal,

como mecánicas dinámicas, esto es, como acciones realizadas por el jugador u otros actores en el videojuego (2014, p. 85). Entonces, las operaciones unitarias en los videojuegos son, pues, toda acción que se suele sintetizar en las llamadas mecánicas de juego o *gameplay*.

La diferencia entre las figuras procedurales y sus operaciones unitarias con respecto a las figuras literarias y sus operaciones unitarias es el entorno en el que se encuentran. Mientras que en los videojuegos están en una representación procedural, en la literatura están en una representación textual, pues, como se ha escrito previamente, solo en la computación tiene lugar la representación procedural. El texto es incapaz de representar proceduralmente. De hecho, se podría sintetizar mejor la diferencia entre la representación procedural y la representación textual de la siguiente manera: mientras que en la representación procedural se *ejecuta* la operación unitaria, de manera que conserva su carácter operacional, en la representación textual solo se *describe* mediante el lenguaje escrito. La ejecución de operaciones unitarias es el principal motivo por el que, para Bogost, la computación tiene un enorme potencial expresivo en comparación con otros medios.

De la conjunción de varias figuras aparecen las formas. En literatura, se cuenta con ejemplos como el soneto o el cuento corto, mientras que en los videojuegos están los motores gráficos o los entornos de trabajo (*frameworks*). Estas formas son significativas cuando se utilizan para crear artefactos expresivos. Bogost (2007) enfatiza que, aunque varios artefactos pueden compartir un mismo fundamento formal, su significado puede variar según el uso que se le dé a este fundamento. Por ejemplo, varios videojuegos que comparten un mismo motor gráfico, si bien son similares dada la presencia de varias figuras procedurales, pueden resultar en artefactos distintos. De la misma manera, un soneto puede servir para expresiones religiosas o amorosas, por ejemplo. Finalmente, del uso conjunto de estas formas, literarias o procedurales, emergen los géneros, como la poesía lírica o la novela de ciencia ficción en literatura, y los géneros de

videojuegos como plataformas, disparos en primera persona (*FPS – First Person Shooter*) o estrategia basada en turnos. Por medio de esta interpretación, Bogost es capaz de explicar las similitudes entre videojuegos y proporciona una base plausible para la categorización de videojuegos basada en géneros.

2.2.4.1.2 Retórica. De acuerdo con Bogost (2007), la retórica se caracteriza por la persuasión y por la expresión. La persuasión indica que se tiene como objetivo cambiar la opinión o el comportamiento de alguien, de modo que se promueva la acción o la inacción, el apoyo o la oposición hacia un fenómeno u opinión determinada. Este cambio de opinión o de comportamiento puede ocurrir en diversos entornos, no solo en el lenguaje. Como ejercicio persuasivo, la retórica puede llevarse a cabo mediante muchas cosas, de ahí que existan dominios como el de la retórica visual, cuyo objeto son las imágenes. A partir de aquí, Bogost (2007) sugiere que, de entre las múltiples cosas con las que se puede persuadir, también están los procesos, los cuales permitirían la apertura del dominio de la retórica procedural.

La retórica como persuasión se trataría de un ejercicio basado en el razonamiento y en la participación del persuadido, en lugar de ser la mera transmisión de ideas u opiniones hacia un receptáculo. Pero, para que este ejercicio tenga lugar, se requiere de un hecho fundamental, a saber, la aceptación de los presupuestos del argumento. Bogost (2007) se apoya en la propuesta de Burke sobre la identificación como el elemento básico de la retórica. La identificación es un requisito para que el ejercicio retórico tenga lugar; es decir, solo cuando el persuadido y el persuasor estén de acuerdo en algunas ideas y nociones básicas alrededor del tópico en cuestión, la retórica, como intento de persuasión, puede ocurrir. En este sentido, la identificación se refiere al acercamiento entre los participantes del ejercicio retórico mediante el consenso de presupuestos comunes.

Por otro lado, la expresión se refiere a que la eficacia de la persuasión requiere de un uso detallado de los elementos utilizados en el ejercicio persuasivo. Se trata de un conocimiento detallado tanto del mensaje que se quiere entregar como de los elementos utilizados para transmitirlo. Existe, pues, una intencionalidad persuasiva inicial, así como un mensaje claro y un dominio del medio con el que se llevará a cabo el intento de persuasión. La expresión también refiere a lo fundamental de la retórica, esto es, la reflexión alrededor del argumento presentado. No se trata de reducir la retórica a la aceptación o a la negación del argumento, sino de estimular a la reflexión alrededor de ese argumento.

La característica de la expresión convierte a los juegos persuasivos en una propuesta de desarrollo de videojuegos muy peculiar, pues la implementación de la retórica procedural requiere de un trabajo minucioso. De hecho, años después de lanzar esta propuesta, Bogost la revisa nuevamente con cierto aire de resignación puesto que, por una parte, se han desarrollado pocos juegos persuasivos desde entonces y, por la otra, su clasificación como juegos persuasivos les hizo ganar interés intelectual, pero poco interés lúdico, de manera que su número de jugadores fue mucho menor al de su número de críticos: «In many cases, particularly when my own games have received attention in the press, it's clear that far fewer people ever thought to play them than thought to think about doing so.» (La Hera et al., 2021, p. 31). Es posible que parte de este resultado se deba a la complejidad de implementar eficazmente la retórica procedural en los videojuegos.

2.2.4.2 Juegos Persuasivos: la Retórica Procedural en la Práctica. Hay, por un lado, una concepción de la retórica que implica persuasión y expresión, esto es, que se propone cambiar la opinión o el comportamiento de alguien de manera eficaz, por el otro, un medio capaz de

representar procedualmente el funcionamiento de sistemas no computacionales. De la conjunción de ambos elementos surge la retórica procedural, que es la aplicación de la retórica en un artefacto procedural. El poder expresivo de los juegos persuasivos consiste en que logran la persuasión mediante el uso concienzudo de su proceduralidad, de manera que persuaden mediante el uso de procesos: «procedural rhetoric address processes, claims or ideas that are expressed with process instead of with speech or with writing or with images or with moving images or with combinations of those media.» (playableUCSC, 2013, 5:26-5:40). En este sentido, un juego persuasivo es un videojuego que representa procedualmente un sistema no computacional, de tal forma que puede sugerir cómo funciona o cómo debería funcionar el sistema representado. Esta persuasión es posible gracias a que los videojuegos, en tanto artefactos operacionales, requieren de alguien más para funcionar: «Games are devices we operate.» (Bogost, 2015, p. 1). Hay, pues, una interacción entre agente externo y videojuego, o lo que es lo mismo, hay interactividad. En lo que sigue se examinan dos implicaciones importantes de la interactividad en los videojuegos en la teoría de Bogost.

2.2.4.2.1 El Acto de Jugar. Por medio de la interacción entre jugador y videojuego se configura el acto de jugar, de modo que el jugador interactúa con el espacio de posibilidades del videojuego, es decir, con los límites de su mundo. En palabras de Bogost: «This is really what we do when we play videogames: we explore the possibility space its rules afford by manipulating the game's controls.» (2007, p. 56). Esta interpretación del acto de jugar es recurrente en los estudios sobre videojuegos. Por ejemplo, en su análisis de *Journey*, Blakey sostiene que el proceso de desarrollo de este videojuego da cuenta de un fenómeno muy interesante para los estudios de ética experimental en videojuegos, a saber, que en el acto de jugar los jugadores pueden, al menos en etapas tempranas de la interacción, desvincularse de su ética personal en pro de la experimentación

con los controles del juego: «in the act of gameplay, real-life personal ethics are to a degree re-prioritised by the interactivity and context of that interactivity in the game world.» (Blakey, 2021, párr. 20). A partir de esta experimentación, los jugadores empezaron a matarse entre sí dentro del juego. Dado que los desarrolladores de *Journey* tenían la intención inicial de generar una experiencia de juego cooperativa y emocional, estos resultados preliminares provocaron una revisión de las mecánicas del juego. Así, deshabilitaron el fuego aliado e incluyeron mecánicas de juego que incentivaban la cooperación.

Al examinar el caso concreto de *Journey*, Blakey se encontró con el problema de la desincronización entre el avatar y el jugador, el mismo al que Gonzalo Frasca se refirió en sus primeros intercambios con la narratología (Frasca, 1999). Frasca retoma la distinción entre *ludus* y *paidia* de Caillios (1986), pero con una ligera precisión, a saber, ambas son formas de juego reguladas, pero *ludus* cuenta con objetivos finales claros, mientras que la *paidia* no. En los videojuegos de tipo *paidia*, es decir, en los videojuegos sin un objetivo final claro, el seguimiento de una estructura narrativa es, de acuerdo con Frasca, más complicado. La razón de esto es sencilla: los jugadores no son autores de una narración, sino meros jugadores, esto es, tienen más interés en la experiencia de juego que en el cumplimiento del rol narrativo de su avatar. En palabras de Frasca (1999):

It is almost impossible to create a puppet of a shy, calm nun and pretend that players will behave according to those traits. If the nun is allowed to control, let's say, a gun, it would be hard that the player doesn't try to kill other people, even if the character was supposed to be a pacifist.

Ya sea parte de un antecedente en los análisis de Frasca o de una versión aplicada en Blakey, lo cierto es que esta interpretación del acto de jugar sugiere un papel activo del jugador. Bogost bebe de esta tesis para sugerir que, en el caso de los juegos persuasivos, la persuasión es un

ejercicio basado en el razonamiento y no en la manipulación. Cabe recordar las principales características de la retórica según Bogost: la persuasión y la expresión. Lo primero se refiere a la capacidad de persuasión para influir en un cambio de opinión o de acción, mientras que lo segundo se refiere a la eficacia de la transmisión de ideas que respalda la persuasión. Para Bogost (2007) hay una clara diferencia entre persuadir y manipular, una diferencia que está marcada en el carácter retórico de la persuasión. Se trata de convencer racionalmente, es decir, con argumentos, en este caso argumentos procedurales, de manera que el individuo persuadido participa de la persuasión. En otros términos, la interactividad, propia del acto de jugar, permite la persuasión como ejercicio retórico.

Ahora bien, los videojuegos pueden, aunque no únicamente, simular sistemas del mundo real de manera muy concreta, esto es, representan sistemas específicos junto a sus respectivos procesos en un intento de imitación. Bogost (2010; 2007) denomina al sistema representado como sistema fuente (*source system*), mientras que denomina como el dominio expresivo del artefacto (*expressive domain of the artifact*) al resultado de su representación específica digital. Cabe precisar que el concepto del dominio expresivo del artefacto no se reduce a los videojuegos como simulación, sino que se aplica, en general, a toda representación procedural en tanto fundamentalmente expresiva, es decir, que transmite ideas, emociones y conceptos al ser creados por mano humana. En el caso de los simuladores, este dominio es el resultado de la visión particular de los desarrolladores del videojuego, quienes posan su mirada sobre sistemas fuente, y de los cuales representan solo lo necesario para sus objetivos. Así, pues, la experiencia de la viveza (*vividness*) a la que se refiere Hill (2004) se encuentra en la capacidad del videojuego de abstraer a los jugadores a una representación procedural específica, más no realista o completamente calcada de la realidad: «But I want to suggest that vividness comes not from immersion, but from

abstraction.» (Bogost & Wright, 2007, p. 45). Así, pues, el acto de jugar evidencia el dominio expresivo del artefacto y permite, asimismo, la persuasión.

2.2.4.2.2 El Rol del Jugador: Completar los Entimemas Procedurales. Los juegos persuasivos pueden persuadir a los jugadores sobre el funcionamiento de los sistemas que representan por medio de la persuasión y de la expresión de la retórica. Ahora bien, en este ejercicio representacional es posible la representación procedural de estructuras argumentativas como los entimemas, ahora entimemas procedurales, con las que el jugador interactúa. Un entimema es un argumento que omite una de sus premisas en favor de una mayor fuerza persuasiva. Esta premisa se suele omitir porque se considera que es racionalmente evidente, es decir, el individuo a persuadir se percata de la premisa omitida mediante el razonamiento. Un ejercicio similar ocurre en los videojuegos con sus entimemas procedurales, los cuales son completados por el jugador mediante sus acciones en el juego, es decir, mediante el acto de jugar. Por ejemplo, en los *Sims* (2000), el jugador comprende que satisfacer las necesidades básicas del personaje (como comer, dormir y socializar) mejorará el bienestar y la felicidad del personaje. El entimema procedural se podría estructurar de la siguiente manera: (i) Satisfacer las necesidades básicas del personaje → (iii) produce un aumento en su bienestar y felicidad. Pero, para lograr este efecto, el jugador debe completar el entimema por medio de (ii) acciones que satisfacen las necesidades del personaje, como alimentarlo, descansar y mantener interacciones sociales positivas, lo que se constituye como la premisa omitida. Entonces, la persuasión se logra mediante el uso eficaz de entimemas procedurales, los cuales son completados, constantemente, por el jugador.

Pero ¿qué ocurre con la subjetividad del jugador una vez que interactúa con el juego persuasivo? Para comprender la relación entre el videojuego y la subjetividad del jugador, Bogost

(2007) sugiere un nombre alternativo para los entimemas procedurales, a saber, brechas de simulación (*simulation gaps*), entendida como el espacio que hay entre el funcionamiento procedural del videojuego y la subjetividad del jugador: «This time, the space between the game's rules and the player's subjectivity is a procedural enthymeme, or what I have called a simulation gap.» (Bogost & Wright, 2007, p. 227). Como era de esperarse, la representación procedural de un sistema fuente en un videojuego puede (y suele) ser distinta de la interpretación subjetiva que tiene el jugador sobre el mismo sistema. La brecha de simulación explicita esta diferencia y acerca el videojuego hacia el contexto personal del jugador, de modo que el videojuego no es un artefacto desvinculado de la subjetividad del jugador: «But for the magic circle to couple with the world, it must not be hermetic; it must have a breach through which the game world and the real world spill over into one another.» (Bogost, 2008c, p. 136). El videojuego no es, entonces, un imperio dentro de un imperio, sino que forma parte de una especie de entramado relacional spinoziano en donde la postura del jugador se encuentra con la postura de los desarrolladores, esta última implícita en la proceduralidad del videojuego. El jugador es quien, al interactuar con el videojuego, es decir, al operar el videojuego y participar de su retórica procedural, es o no persuadido. Si se sigue con el ejemplo de los *Sims* (2000), es el jugador quien decide o no aceptar el argumento según el cual (i) satisfacer las necesidades básicas del personaje, (ii) por medio de acciones como la alimentación o el descanso, (iii) aumentará sus niveles de bienestar y de felicidad, o si, por ejemplo, considera que algunas de estas acciones no son realmente necesarias para el bienestar o, tal vez, discrepa con el argumento a un nivel más profundo en tanto que su interpretación de la felicidad difiere de la presentada por el videojuego. Esta reflexión extrae el argumento a espacios de discusión externos al videojuego, como foros de *Reddit* o servidores de *Discord*.

Una vez se lleva a cabo la interacción y el intento de persuasión del videojuego, la brecha de simulación puede provocar, en caso de no lograr la persuasión, lo que Bogost denomina fiebre de simulación (*simulación fever*), que es la respuesta subjetiva del jugador a la brecha de simulación. La fiebre de simulación tiene dos características fundamentales: primero, no es propia del videojuego, sino que es el resultado de la interacción entre el videojuego y el jugador: «the dialectic between unit operations and subjectivity that constitutes simulation fever is extrinsic, not intrinsic, to the game.» (Bogost, 2008c, p. 133). Es decir, mientras que la brecha de simulación es parte del videojuego, la fiebre de simulación es posterior a la interacción y, por lo tanto, extrínseca al videojuego. Segundo, se trata de una condición subjetiva de carácter cognitivo, pues varía entre jugadores de acuerdo, por un lado, con la interpretación subjetiva de cada jugador sobre el sistema representado y, por el otro, con la manera como cada jugador le hace frente a esta diferencia, como los dos casos en desacuerdo que se mencionaron atrás en el ejemplo de los *Sims*. En palabras de Bogost (2007): «This discomfort is what I have called simulation fever, an internal crisis wrought between the game's rules and the player's subjective response to them.» (p. 249). Entonces, el jugador que interactúa con el videojuego debe, constantemente, completar estos entimemas, ejercicio que no es más que el intento del videojuego por persuadirlo, y que puede, asimismo, provocar la fiebre de simulación.

Hasta el momento queda claro que Bogost sostiene que los juegos persuasivos, como videojuegos imbuidos de retórica procedural, fomentan la reflexión alrededor de aseveraciones, hechas por los desarrolladores, sobre sistemas específicos. Los juegos persuasivos provocan una reflexión sobre el sistema representado, hecho que no había tenido lugar antes de la interacción y que, potencialmente, puede producir una fiebre de simulación. Ahora bien, a pesar de que, a simple vista, la fiebre de simulación parece un efecto completamente distinto de lo esperado en el ejercicio

persuasivo, en realidad se trata, más bien, de un efecto propio de la retórica procedural. En una lectura superficial se considera que en el ejercicio retórico solo hay dos resultados posibles: persuadir o no persuadir. En el primer caso, el jugador es persuadido por la representación del videojuego y adopta su visión particular sobre el sistema representado. En el segundo caso, cuando el jugador no es persuadido, se asume que la persuasión ha fallado. No obstante, para Bogost (2008b) existen, en realidad, tres caminos, todos igualmente válidos para la retórica procedural, los cuales son: aceptar el argumento, cuestionarlo o rechazarlo. En los dos últimos casos surge la posibilidad de la fiebre de simulación, en la medida en que es una respuesta subjetiva que emerge cuando el jugador no es completamente persuadido por la representación específica del sistema fuente. Y, en tanto que se trata de un ejercicio racional que requiere de la habilidad del jugador tanto de comprender la representación del videojuego como de distinguirla de su propia interpretación, en los tres casos hay reflexión. En palabras de Bogost: «Persuasion is related to the player's ability to see and understand the simulation author's implicit or explicit claims about the logic of the situation represented.» (2007, p. 333). En este sentido, se requiere de la habilidad cognitiva del jugador como elemento fundamental, de modo que aceptar, cuestionar o rechazar el argumento parten de un elemento en común y son, por lo tanto, resultados igualmente deseables. La fiebre de simulación es, al igual que el convencimiento, un resultado posible de la retórica procedural. Todo lo anterior resalta, una vez más, la perspectiva de Bogost con respecto a la persuasión como radicalmente distinta de la manipulación, motivo por el cual se distancia de propuestas como la *Tecnología Persuasiva* de Fogg (2007, pp. 59-62), en la medida en que es más cercana a la manipulación que a la persuasión.

2.3 Aspectos generales de la teoría de Bogost

En el presente capítulo se revisaron dos modelos de análisis de videojuegos de Bogost, a saber, los estudios de plataformas y la proceduralidad. Se ha propuesto un recorrido general de estos dos modelos, con énfasis en las potenciales respuestas a la pregunta ¿Qué es un videojuego? Del recorrido se destacan dos aspectos clave. Primero, se evidencia el continuo interés de Bogost por una visión de los videojuegos que integre la computación y la cultura. Los videojuegos no son círculos mágicos; antes bien, inciden, de una u otra forma, en los jugadores y, en última instancia, en la sociedad que los vio nacer. Segundo, los videojuegos llevan consigo lo que se puede denominar intencionalidad inicial. Ya sea para desarrollar un videojuego para una plataforma distinta, ya sea para persuadir o incitar a la reflexión, lo cierto es que esta intencionalidad inicial requiere del uso minucioso de los elementos disponibles. La creatividad es un componente esencial en el proceso de desarrollo de videojuegos, y se ve moldeada tanto por la intencionalidad inicial como por los elementos tecnológicos disponibles. Finalmente, los videojuegos son la combinación de una multitud de componentes (procesos/operaciones unitarias, intencionalidad inicial, influencias culturales, tecnologías disponibles), lo que da cuenta de su complejidad. Esta estructura argumentativa será revisada desde la perspectiva teórica de la Ontología Serial del Software en el siguiente capítulo.

3. La teoría de Bogost a la luz de la Ontología Serial del Software

Este capítulo se dedica a la revisión crítica de la teoría de Bogost desde dos secciones mutuamente complementarias. La primera sección presenta un examen de los dos métodos de análisis de Bogost a partir del marco conceptual de la Ontología Serial del Software. La segunda presenta una interpretación crítica de Bogost alrededor de algunas objeciones a su teoría.

Es necesario precisar que, si bien existe cierta afinidad conceptual entre el objeto de estudio y el método de análisis, no se trata de forzar a la teoría de Bogost a encajar dentro de los parámetros de la Ontología Serial del Software. En la medida en que se quiere destacar cómo la propuesta teórica de Bogost proporciona una perspectiva valiosa para comprender los videojuegos como software, no se pierden de vista los límites de esta afinidad conceptual.

3.1 Revisión de los dos métodos de análisis

De acuerdo con lo revisado en el segundo capítulo, en la teoría de Bogost hay dos métodos de análisis de videojuegos, a saber, el análisis de plataformas y la proceduralidad. Si bien el análisis de plataformas fue aplicado, particularmente, a la Atari VCS, se trata de un método de análisis con gran apertura para el estudio de otras plataformas. El fundamento teórico de este método es que el desarrollo y la recepción de los videojuegos está íntimamente conectado con la plataforma que lo soporta, la cual está, a su vez, conectada con el contexto tecnológico y cultural del momento. Pero no se trata de una conexión trivial, como si la tecnología y la cultura tuviesen una influencia secundaria. Antes bien, los dos elementos resultan determinantes en la producción de la

plataforma, y por lo mismo, en el desarrollo y la recepción de los videojuegos de dicha plataforma. De esta manera, la propuesta de Bogost sostiene que el estudio de los videojuegos implica un análisis técnico y detallado del soporte tecnológico, lo que implica, a su vez, una consideración de la tecnología y de la cultura que dio lugar a ese soporte tecnológico.

Los videojuegos producidos para la Atari VCS fueron, en su gran mayoría, cuidadosamente desarrollados, toda vez que durante este proceso se tuvieron en cuenta las capacidades técnicas de la plataforma con el mismo nivel de importancia que se tuvieron en cuenta las intenciones de los desarrolladores. De hecho, como se sugirió en el capítulo anterior, las capacidades técnicas determinaron la aplicación de las intenciones de los desarrolladores, mientras que estas intenciones determinaron, a su vez, el uso de las capacidades técnicas de la plataforma. De acuerdo con Bogost (2009), es a partir de este intercambio durante el desarrollo de videojuegos para la Atari VCS que se dio paso a formas de expresión computacional que, con el tiempo, se instalarían en el mundo del desarrollo de videojuegos. La generación de un espacio de juego más allá de la pantalla en *Adventure* (1979), los primeros pasos en el implementación de IA en *NPCs* en *Video Olympics* (1977) o en *Video Chess* (1979), entre otros, son ejemplos de ideas significativas que marcarían el desarrollo de videojuegos venidero. En este sentido, el estudio particular de cada videojuego comprende un análisis de su contexto tecnológico y cultural. En palabras de Bogost: «Each video game is a particular piece of software created and run on a particular piece of computer hardware at a particular moment in time.» (CU Boulder Libraries, 2011, 00:14:17-00:14:36).

Desde la perspectiva de la Ontología Serial del Software, este primer método de análisis constituye una visión de los videojuegos que enfatiza en su carácter material. Los videojuegos, al igual que cualquier otro software, no son solo código; también envuelven el llamado hardware codificado o, en los términos de este trabajo, el sustrato material. No obstante, es necesario

establecer que el método de análisis de los estudios de plataformas no forma parte, propiamente, de una lectura ontológica de los videojuegos. Esto se hace evidente en la medida en que Bogost se refiere a las capacidades técnicas de una plataforma como restricciones o limitaciones técnicas: «the constraints of the Atari VCS» (2009, p. 38). Para la Ontología Serial del Software no se trata de limitaciones técnicas, sino de que las funcionalidades de lo digital se encuentran circunscritas, ontológicamente, en los componentes físicos donde lo digital tiene su realización física. Al mismo tiempo, las intencionalidades particulares de un computador digital se encuentran circunscritas, ontológicamente, en su intencionalidad general. Se trata de una especie de conexión ontológica entre un universal y sus particulares. Entonces, mientras que Bogost se centra en las restricciones técnicas prácticas que un desarrollador de videojuegos pueda encontrar en una plataforma específica, para la Ontología Serial del Software el *quid* del asunto radica en la pregunta por lo esencial y definitorio para el software, en donde, *grosso modo*, los particulares están dentro de lo general en términos ontológicos. Con todo, la visión de Bogost se puede incluir dentro de esta ontología como una manera similar, aunque menos precisa, de señalar la importancia de una comprensión del soporte tecnológico en los estudios sobre videojuegos.

Sobre el método de análisis de la proceduralidad cabe destacar dos cosas, a saber, su carácter simbólico, por un lado, y el lugar que ocupa la intencionalidad y la interacción en la teoría de Bogost, por el otro. Si el análisis de plataformas representa una preocupación por el carácter material de los videojuegos, la proceduralidad representa un interés por su carácter simbólico. La proceduralidad, y más precisamente la representación procedural, es de carácter simbólico en la medida en que permite la representación de sistemas conceptuales, es decir, de sistemas intangibles y abstractos. Como resultado, también es posible, mediante la aplicación de la retórica procedural, elaborar argumentos sobre sistemas conceptuales. En sus propias palabras: «But since

procedurality is a symbolic medium rather than a material one, procedural rhetorics can also make arguments about conceptual systems» (Bogost, 2008b, p. 126). Además, es necesario reiterar que, para Bogost (2008b; 2007), el acto de jugar consiste en la exploración de un espacio de posibilidades determinado por las reglas del juego. Este espacio comprende multiplicidad de elementos simbólicos en tanto que está construido en código, el cual es entendido aquí como puramente simbólico. En este sentido, mientras que los sistemas fuente pueden ser tanto tangibles como conceptuales, el dominio expresivo de un videojuego, al estar construido en código, es esencialmente simbólico. Al ser simbólica, la representación procedural puede representar sistemas de cualquier tipo sin discriminación de su naturaleza.

Por otro lado, Bogost distingue entre la intencionalidad de los desarrolladores y la interactividad de los usuarios. Para Bogost, «no video game is produced in a cultural vacuum» (2008b, p. 128). Esto es, los videojuegos son artefactos, pues son creados por seres humanos, pero también son culturales en la medida en que estos mismos creadores transmiten sus ideas, sus perspectivas, en fin, sus prejuicios humanos, a los videojuegos. Aquí es donde tiene lugar el dominio expresivo del videojuego, el cual es construido y moldeado por los desarrolladores de acuerdo con sus intereses particulares. Entendido así, la retórica procedural es una manera más controlada y consciente de llevar a cabo esta transmisión mediante el uso eficaz de la proceduralidad del videojuego.

En cuanto a la interactividad del usuario, Bogost (2015) sostiene que, dado que los videojuegos son dispositivos, requieren de la acción humana para funcionar. Para ilustrar esta idea, Bogost describe al videojuego como una máquina defectuosa, de modo que el papel del jugador es completar el vacío creado por la falla: «a game is like a broken machine, and the thing that you do as a player is fill in for the broken part. You're there to make it operate—if it's operating well, then

there's no need for you» (Williams, 2019, p. 477). Sin la existencia de esta falla que solicita a un ente externo para completarse, la presencia humana sería innecesaria. Asimismo, la interacción del jugador con las reglas del videojuego es esencial para explorar el dominio expresivo del videojuego, lo cual es un requisito fundamental para la persuasión en los juegos persuasivos, como se explicó en el capítulo anterior con respecto a los entimemas procedurales.

Ahora bien, aunque la Ontología Serial del Software y la teoría de Bogost utilizan terminología ligeramente diferente, comparten un enfoque teórico similar. En la Ontología Serial del Software, la intencionalidad hacia arriba implica que el usuario *concretiza* el propósito específico del programa mediante el uso de extensiones del mismo programa. La propuesta de Bogost se traduce en que el jugador *completa* los entimemas procedurales mediante su interacción con el videojuego. En la primera se habla de *intencionalidad* y de *concretizar*, mientras que en la segunda se habla de *interactividad* y de *completar*. Sin embargo, desde una lectura de la Ontología Serial del Software, la propuesta de Bogost bien puede referirse a la intencionalidad a nivel de usuario en tanto que explica cómo este concretiza y da significado al programa mediante acciones específicas dentro del contexto del videojuego. Estas acciones están circunscritas en las mecánicas de juego, lo que confina cualquier acción del jugador dentro de los parámetros del propósito específico del software, esto es, dentro del mundo regulado del videojuego. Es decir, a nivel ontológico, toda acción que el jugador pueda llevar a cabo en un videojuego ya se encuentra circunscrita en el mundo regulado del videojuego, o lo que es lo mismo, *es posible*. Si un entimema procedural se puede completar de *x* manera, es porque es posible ser completado de esa manera dadas las mecánicas del videojuego.

Con todo lo anterior no se pretende argumentar que tanto la Ontología Serial del Software como la teoría de Bogost de la proceduralidad reducen el rol del usuario a mera interacción final,

como si el videojuego fuese una especie de relato inconcluso que solo requiere de una acción específica del usuario para concluir. Quizá la metáfora de Bogost sobre los videojuegos como máquinas dañadas no haya sido la más acertada. Pero lo cierto es que la intencionalidad a nivel de usuario es más compleja, pues los caminos posibles que el usuario puede tomar, si bien deben estar estrechamente vinculados con el software, son heterogéneos. Una analogía con los objetos técnicos puede ayudar a comprender mejor esta tesis (Maldonado, 2019): un objeto técnico x fue construido de tal manera que cumpla la función y . No obstante, a nivel de usuario, los usos del objeto técnico pueden variar por la combinación de dos aspectos: primero, el saber-cómo del agente usuario, pues es al usuario a quien se le ocurre usar el objeto técnico de una manera no-estándar, es decir, el usuario posee cierta habilidad práctica para usar el objeto técnico de maneras alternativas a la función estándar. Segundo, el entendimiento, por parte del agente humano, de las propiedades (en este caso físicas) del objeto, esto es, comprende cómo y por qué puede usar el objeto de la manera como pretende usarlo. Así, el objeto técnico x con función estándar y puede ser usado de manera p porque tanto las propiedades físicas del objeto como la habilidad y el conocimiento del usuario así lo permiten. A las variadas posibilidades de uso de un objeto gracias a sus propiedades físicas se le conoce como ambivalencia tecnológica (Maldonado, 2019). Nótese, de nuevo, que la intencionalidad a nivel de usuario se encuentra circunscrita en el propósito específico del objeto, en este caso, del software. Si, por ejemplo, un arma de fuego es usada como un martillo, es porque tanto la habilidad y el conocimiento del usuario sobre el objeto, como las propiedades físicas del objeto mismo, pueden dar lugar a este uso particular. No se trata del uso estándar del objeto, pero, dadas sus propiedades físicas, este uso particular se circunscribe en el propósito específico del objeto. A esta multiplicidad de intencionalidades a nivel de usuario se le denomina ambivalencia intencional (Maldonado, 2019). De este modo, primero tiene lugar la ambivalencia tecnológica del

objeto, la cual permite, posteriormente, la ambivalencia intencional, y de esta última se derivan otros niveles de ambivalencia, como la moral y la legal².

Piénsese, por ejemplo, en un programa de edición de textos. La producción literaria que se puede generar en él es bastante amplia. Desde ensayos, novelas, cuentos cortos, textos con letra muy pequeña, textos con letra cursiva y al revés. Todo depende tanto de las propiedades del programa como de la habilidad práctica del usuario. Como ejemplos de ambivalencia intencional en videojuegos se puede citar la superación de un obstáculo de diferentes maneras, o descubrir y hacer uso de *exploits* que le permitan al jugador omitir secciones del videojuego, entre otros. Solo por nombrar un ejemplo concreto, en *Left 4 Dead 2* (2009), un videojuego de disparos en primera persona, existe la mecánica del empuje. Cuando el jugador ejecuta esta acción, el personaje usa su arma equipada para empujar. Los jugadores descubrieron que, con las escopetas semiautomáticas, las cuales requieren del desplazamiento hacia atrás de la corredera para cargar cada cartucho antes de poder dispararlo, se puede hacer uso de la mecánica del empuje para omitir la animación de la corredera, mientras se conservan sus mismos efectos, esto es, preparar el siguiente cartucho para disparar. De esta manera, y dado que la animación de empujar es más rápida que la animación de la corredera, la frecuencia de disparo con este tipo de armas se vuelve bastante similar en velocidad a la de sus contrapartes automáticas. Se trata de unos usuarios del software que poseen habilidad práctica y conocen muy bien el videojuego, su objeto. En otros términos, los jugadores conocen y entienden las mecánicas del videojuego, y poseen la habilidad necesaria para llevar a cabo, con rapidez y precisión, las dos acciones en sucesión (disparo-empuje). Acciones de este tipo no fueron

² Es por esto que, desde la Ontología Serial del Software (Maldonado, 2019), se entiende que lo digital no debería ser juzgado bajo los mismos parámetros morales y legales que se usan en el mundo analógico. Sin embargo, esto no significa que los programas estén desprovistos de juicio moral, sino que, intrínsecamente, albergan diversas opciones morales en su aplicación. Dada esta misma diversidad de opciones, no se les puede restringir y juzgar bajo un único parámetro de evaluación moral.

previstas por los desarrolladores, pero, dadas las mecánicas del videojuego, son acciones *posibles* dentro de su marco de intencionalidades.

Ahora bien, la analogía con el objeto técnico plantea ciertas dificultades teóricas. La primera y más evidente es que el software no es un objeto, sino un proceso. La dimensión estructural del software propone que la existencia del software comprende el funcionamiento de todas sus series al unísono. La segunda dificultad es que la argumentación simondoniana sobre la cual se fundamenta la analogía con el objeto técnico resulta insuficiente para explicar las capas de intencionalidad en el software. Esto se debe a dos presupuestos teóricos: el primero es que supone la existencia de un *telos* original para el objeto técnico, el segundo es que el concepto de hipertelia, con el cual Simondon (2008) explica la aparición de múltiples *telos* en un objeto técnico, no es equivalente al fenómeno de la sobrecodificación de la Ontología Serial del Software. Por un lado, resulta complicado sugerir un *telos* original en un software particular y adherirse a él de la misma manera que al *telos* original de un objeto técnico, y por el otro, aplicar el concepto de hipertelia al software para dar cuenta de sus capas de intencionalidad supone enfrentar el mismo inconveniente con respecto a los objetos técnicos, a saber, dada la acumulación de *telos*, se desdibuja el *telos* original, y con ello, «el objeto se ve desestabilizado, generando un funcionamiento indeseado» (Gavarini, 2015, p. 20). En este sentido, la multiplicidad de intencionalidades se podría interpretar de manera peyorativa, y el software no tendría tanta diferencia con respecto a un objeto técnico.

A pesar de los inconvenientes, la analogía entre el software y el objeto técnico resulta útil para entender el fenómeno de la ambivalencia intencional en el software. Sin embargo, no debe ser tomada de manera estricta. Aunque el software es un proceso, es plausible reconocer la existencia de un *telos* original, al menos en la forma de un planteamiento inicial. No obstante, dada la multiplicidad de capas de intencionalidad que se superponen desde su génesis, este *telos* no

resulta tan relevante. Aquello que, con Bogost, se categorizó como la intencionalidad inicial y que bien podría tomarse como el *telos* original, tan solo sería uno de los múltiples propósitos que coexisten en la génesis de un videojuego. Esta intencionalidad inicial solo abarca, por así decirlo, al *telos* de los desarrolladores del videojuego, excluyendo las diversas intencionalidades presentes en el sustrato material, el motor gráfico del videojuego o el lenguaje de programación, entre otros. En este sentido, se propone un término alternativo para conceptualizar el fenómeno de la sobrecodificación en el software: el *telos* semi-abierto. Al emplear este término, se sugiere una mayor complejidad y flexibilidad en la intencionalidad del software, de modo que se reconocen los múltiples propósitos que implica, especialmente aquellos que emergen en el nivel del usuario. La distinción entre la hipertelia del objeto técnico y el *telos* semi-abierto en el software radica en que, mientras que en el primero la ambivalencia intencional surge debido a una sobrecarga de *telos* que, en última instancia, desvía del *telos* original, algo que, como se ha indicado, puede tomarse de manera peyorativa, en el segundo, la ambivalencia intencional aparece precisamente porque el *telos* original pierde relevancia con el tiempo. El *telos* semi-abierto supone la consideración de todas las intencionalidades; la posibilidad de concretizar un software de una manera específica es factible porque esa concretización se encuentra dentro de las diversas intencionalidades implicadas por el *telos* semi-abierto del software.

La intencionalidad a nivel de usuario no se refiere, pues, al acto de completar un trabajo incompleto mediante una acción específica. Esto se debe a que, primero, la manera en la que se completaría este trabajo no es una sola, y por lo mismo, tampoco suele ser mediante la opción que se pensó inicialmente. En términos de Simondon (2008) el objeto técnico cuenta con un *telos* original, pero puede tener varios y devenir en hipertelia. En cambio, el software, al ser más bien un proceso con capas de intencionalidades, cuenta con *telos* semi-abierto. Asimismo, la

concretización de un software depende de sus propiedades y del conocimiento y habilidades de su usuario, lo cual varía. De manera similar a la individuación en Simondon (2008), un programa nunca está realmente completo; lo que ocurre es una concretización con un usuario y circunstancias particulares. Es en esta parte de la argumentación donde aparece tanto la clasificación tentativa de los distintos niveles de usuario como la de los niveles de ambivalencias. Segundo, porque se trataría de una reducción de la intencionalidad, quizá demasiado afín a la interpretación de la interactividad de Sarah Roberts (Como se citó en Penny, 1996) quien sostiene que la interacción está cargada de cierto aire de ilusión, pues supone que el artista (en este caso, el desarrollador) comparte con el espectador (en este caso, el jugador) la elección de escoger, pero, en realidad, el artista ya ha planificado cada opción posible. Se trata de una reducción de la intencionalidad porque no tiene en cuenta las diversas capas de intencionalidad superpuestas entre sí, y le da demasiado énfasis al desarrollador, cuando, como se exploró en el primer capítulo, incluso el usuario-programador, quien despliega, mediante la modificación del código, el propósito específico del software, lo hace de acuerdo con las capas de intencionalidad que le preceden. Y esto incluye, a partir de una interpretación de la teoría de Bogost desde la Ontología Serial del Software, a las capas de intencionalidad que ya están presentes en el sustrato material que es la plataforma del videojuego.

Se concluye esta sección con lo siguiente. Bajo la interpretación de los dos métodos de análisis de videojuegos se hace evidente que el filósofo anglosajón mantiene la distinción entre hardware y software, cosa que no resulta sorprendente, dada la tradición filosófica con la que dio forma a su obra. Pero, es destacable que cada método de análisis atiende a un aspecto importante de los videojuegos. En particular, el método de análisis de la proceduralidad presta atención a los videojuegos como esencialmente código, de modo que se trataría de una apuesta teórica que le

apunta a su carácter digital. Si bien, en principio, Bogost (2009) sugirió que el origen del término *videojuego* se debió a algo tan trivial como la presencia de una pantalla, mediante este nuevo acercamiento se estudian los videojuegos como software (entendido solo como código), esto es, como tecnología digital. La propuesta de la retórica procedural, derivada de la teoría de la proceduralidad, es el intento por ofrecer un marco conceptual de desarrollo de videojuegos que no solo tenga presente su carácter digital, sino uno mediante cual se intente extraer el mayor provecho posible. Esta idea se desarrollará con más detalle en la siguiente sección. Por lo pronto, considérese que, por un lado, se mantiene la distinción entre hardware y software, y por el otro, se mantiene la interpretación del software como código, y este, a su vez, como puramente simbólico. No obstante, una lectura en conjunto de ambos métodos de análisis revela una aproximación más coherente y completa que si se les explora por separado, y la teoría de la proceduralidad revela que, en el fondo, subyace una preocupación por el carácter digital de los videojuegos que se encauza con los propósitos de la Ontología Serial del Software.

3.2 La especificidad de los videojuegos

Se propone iniciar esta sección con un breve repaso de algunos de los puntos clave de la corriente ludológica. Aarseth (1997), uno de los primeros autores de esta corriente, propuso una interpretación de los videojuegos como textos ergódicos, es decir, como artefactos que exigen del esfuerzo físico del jugador para ser completados. Lo importante en el estudio de los videojuegos es el videojuego en sí mismo, todo lo demás es secundario. La ludología se distinguiría de la narratología en tanto que aquella presta atención al videojuego, mientras que esta última presta atención a su narrativa. No obstante, esto no implica que, para los ludólogos, la narrativa sea

excluida por completo. Para Gonzalo Frasca, por ejemplo, los videojuegos, en tanto simulaciones, no son narración en sí mismos, pero sí pueden contener narraciones: «But the simulation itself is something bigger than narrative. It is a dynamic system that yes, contains thousands of potential "stories", but it is larger than the sum of its parts» (Frasca, 2001). Años más tarde, el debate tomaría una nueva forma gracias a la distinción, ofrecida por Juul (2016), entre dos perspectivas del jugar (*game playing*), a saber, una centrada en el jugador (*player-centric*) y otra centrada en el videojuego (*game-centric*). La idea detrás de esta nueva lectura ya no sería el videojuego en sí mismo y sus características, sino la experiencia misma del jugar. La perspectiva centrada en el jugador considera el jugar como una actividad fundamentalmente libre, mientras que la centrada en el videojuego la considera como una actividad determinada por las reglas del videojuego. Así, Juul desarrolla su investigación bajo la pregunta «Is game playing a free activity, or is it determined and controlled by the game rules?» (2016, p. 351). Este repaso resulta útil para analizar algunas de las lecturas alrededor de la propuesta de Bogost, antes de presentar una perspectiva adicional.

Para Michael Sicart (2011), la teoría de la proceduralidad de los ludólogos en general, y en particular la de Bogost, sostiene que el sentido de los videojuegos se encuentra dentro de sus reglas. En tanto que los videojuegos están compuestos, fundamentalmente, por reglas, entonces «the game is the rules» (Sicart, 2011, párr. 22). Esto simplifica el acto de jugar y el papel del jugador en el videojuego. Jugar se reduce al descubrimiento del sentido implícito en las reglas del videojuego mediante la interacción del jugador con esas mismas reglas durante el acto de jugar. Esto explicaría por qué la propuesta de los juegos persuasivos se mantiene dentro de los límites de los videojuegos para un solo jugador, en tanto que la multiplicidad de jugadores en videojuegos en línea dificulta la transmisión del sentido del videojuego mediante el acto de jugar (Sicart, 2011). En efecto, varios estudios apuntan a que la presencia de otros jugadores se traduce en un elemento más de la

experiencia de juego, lo que genera resultados significativos y diferentes a experiencias de un jugador (Acevedo-Merlano & Chauz Lizarazo, 2016; Grimes & Feenberg, 2009; Kulovitz, 2013). En resumen, Sicart (2011) considera la propuesta de Bogost como una lectura centrada en el desarrollador (*designer-centric*).

Que la proceduralidad de Bogost sea una lectura centrada en el desarrollador significa que no se trata de una interpretación del videojuego en sí, como lo sería la postura *game-centric*, sino una que ve en el videojuego una herramienta útil para la transmisión de mensajes, al jugador como el mero receptor del mensaje, y al acto de jugar como el elemento intermedio que permite la transmisión. Según esta lectura, el concepto de la fiebre de simulación es insuficiente para dar cuenta de la experiencia de juego y del papel del jugador (Sicart, 2011).

En principio, la crítica de Sicart resulta plausible, pero también resulta válida la respuesta de Mark Nelson (2012). En primer lugar, para Nelson, Sicart se opone a una forma de transmisión de mensajes que hace uso de una interpretación rígida de la retórica. A esta forma de transmisión Nelson (2012) la denomina mano dura o autoritaria (*heavy-handed*). Se trataría, pues, de una interpretación de la retórica que no se restringe al terreno de la proceduralidad; la transmisión de mensajes mediante el uso rígido de la retórica, de tal forma que fallar en la transmisión del mensaje se le considere un fracaso, es algo común en otros dominios (Nelson, 2012). Entonces, Sicart no se opone, realmente, a la proceduralidad ni a su uso retórico, sino a una forma de retórica parecida a la propaganda. En segundo lugar, de acuerdo con Nelson (2012), la crítica de Sicart tampoco deja muy claro si la acusación a la teoría de Bogost como una propuesta centrada en el desarrollador se debe a que, para Sicart, las reglas no contienen, en lo absoluto, el sentido del videojuego, o si, en realidad, sí lo hacen, pero en una menor medida. Tras una lectura rápida de la

producción literaria de Sicart sobre el tema, Nelson (2012) concluye que se trataría de la segunda opción. En sus palabras:

I read him as objecting to the type of ulterior motive: no to rhetoric, propaganda, encoding an opinion, but yes to intervention, and setting up situations, with meaning not conveyed, but jointly produced from the design and the players' interaction with the design. (Nelson, 2012, párr. 16)

A pesar de las objeciones planteadas a la crítica de Sicart, Nelson concluye de una manera que le otorga cierta validez. Nelson (2012) sugiere que, si bien la retórica es un uso válido de la proceduralidad, no la considera particularmente útil. En su lugar, propone un acercamiento más performativo: «Procedurality should be more performative than rhetorical» (Nelson, 2012, párr. 19). Esta propuesta es interesante porque plantea dos lecturas: o bien sugiere que la retórica no es intrínsecamente performativa, lo cual es discutible ya que la retórica se centra en lograr efectos con el discurso, lo cual es una forma de performatividad, o bien propone que, aunque ya es performativa, la retórica podría potenciar aún más esta característica. En todo caso, la pregunta que surge es por qué tanto Sicart como Nelson tienen reservas sobre la unión entre retórica y proceduralidad. Todo parece indicar que estos reparos se remontan a la idea de Bogost sobre los videojuegos como máquinas dañadas que solicitan de un ente externo para funcionar. Como se argumentó en la sección anterior, esta interpretación de los videojuegos da a entender, inicialmente, que jugar un videojuego consiste en completar un relato inconcluso mediante una acción específica y premeditada por los creadores del relato. Pero esto se trataría de una malinterpretación de la propuesta de Bogost, una según la cual el filósofo anglosajón reduce la intencionalidad a nivel de usuario a un único camino posible, lo que se constituiría, a su vez, como una interpretación rígida de la retórica. En vez de eso, la propuesta de Bogost da cuenta, en concordancia con la Ontología

Serial del Software, de la amplitud de la intencionalidad a nivel de usuario, en tanto que el software posee *telos* semi-abierto. Se desarrollará este argumento en cuatro puntos: (i) la retórica procedural como propuesta de desarrollo de videojuegos, (ii) la importancia del usuario en la transmisión del mensaje, (iii) la representación procedural como la especificidad de los videojuegos y (iv) las funcionalidades de los videojuegos en tanto que digitales. En lo que sigue se revisará cada punto en ese mismo orden.

En primer lugar, se insiste en que la retórica procedural es una propuesta de desarrollo de videojuegos, de manera que debe tomarse como tal. Se trata de una propuesta que atiende a lo característico del videojuego, a saber, su proceduralidad, o más precisamente, a su capacidad de representar proceduralmente, para una mejor transmisión de mensajes de desarrollador a jugador. No se trata, pues, de una reducción del papel del usuario, sino de prestar cierta atención al proceso de desarrollo del videojuego para llevar a cabo la transmisión de mensajes con mayor eficacia. En tanto que el desarrollador se propone crear un juego persuasivo con un mensaje particular, es más consciente del mensaje y de la manera como lo trata de comunicar en su videojuego. En este sentido, y dado que no se puede pensar en los videojuegos sin tener en cuenta la mano humana que los creó, se le apunta a un mayor nivel de consciencia por parte del desarrollador sobre su papel e influencia en el videojuego, sin menoscabar la importancia del jugador, como se verá en el segundo punto.

En segundo lugar, con la retórica procedural Bogost presta especial atención al jugador. En términos de Bogost (2007), en un juego persuasivo es el jugador quien debe contar con la habilidad cognitiva suficiente tanto para entender los argumentos del videojuego sobre el sistema fuente como para contrastarlos con su propia visión sobre ese mismo sistema. La retórica es, como se ha insistido, un ejercicio racional, de modo que se requiere de la contribución de todos los

involucrados. En términos de la Ontología Serial del Software, el jugador debe contar con la habilidad y el conocimiento necesarios para completar los entimemas procedurales y así dar lugar al intento de persuasión. Una interpretación así no ignora al usuario, al contrario, lo tiene en cuenta de tal manera que el mensaje le llegue con más claridad. ¿Cómo se podría formular una manera efectiva de transmitir un mensaje si no se tiene en cuenta a su receptor?

Asimismo, es el usuario y su capacidad simbólica de sobre-codificar lo que permite el intento de persuasión en primer lugar. En efecto, una imagen en una pantalla se sobre-codifica en un personaje de un videojuego, es decir, un montón de píxeles cuentan como un personaje, cuyo rol, a su vez, es asumido por el usuario en el contexto del videojuego. De hecho, los entimemas procedurales son acciones que completa el jugador mediante esta misma capacidad simbólica: una acción *A* llevada a cabo por el jugador a nivel físico cuenta como una acción dentro del juego, la cual desencadena unas consecuencias. He ahí la estructura básica de un entimema procedural, el cual requiere de la sobre-codificación del usuario para completarse. Pero, la importancia del usuario no se agota aquí. La ejecución de acciones por parte del usuario, o lo que es lo mismo, *jugar*, constituye el consenso entre el jugador y el desarrollador en el fenómeno de la identificación de Burke (1969). Como afirma Petrowicz (2014), un videojuego cuenta con unos presupuestos que el jugador debe *aceptar e implementar*, mediante el uso de las mecánicas del juego, para poder introducirse en su mundo, o en este caso, en su dominio expresivo. Dicho en sus propios términos: «El usuario no solo acepta la ideología necesaria para completar la brecha de simulación, sino que también la implementa con sus acciones.» (Petrowicz, 2014, p. 88). *Aceptar* los presupuestos, el cual es un requisito fundamental para dar inicio al acto de jugar, es retórico, mientras que *implementar* los presupuestos, lo cual se hace mediante el uso de las mecánicas reguladas del videojuego en el acto de jugar, es procedural. Así, se consigue una especie de diálogo continuo

entre el jugador y el desarrollador, en donde este último es consciente de su mensaje y de la manera como lo transmite, mientras que aquel es capaz de recibir y de entender el mensaje sin mayor inconveniente. No obstante, el resultado del intercambio queda en manos del jugador, lo que da lugar a múltiples *telos* en el nivel de usuario: si acepta, cuestiona o rechaza el argumento, y de qué manera lo hace. Todo lo anterior se debe a la capacidad de representar procedualmente de los videojuegos, lo cual constituiría su especificidad, como se verá en el siguiente punto.

En tercer lugar, aunque la proceduralidad extiende su dominio más allá de los videojuegos, Bogost concede que la representación procedural les es propia. La proceduralidad es la noción fundamental de la autoría de procesos (Bogost & Wright, 2007), de manera que se trata de una lectura ontológica de cualquier sistema, sean mecánicos, conceptuales, culturales, sociales, históricos, etc., como formados por operaciones unitarias. El análisis unitario y la crítica comparativa son útiles para la comprensión de cualquiera de estos sistemas a partir del concepto de operación unitaria, ya sea mediante un examen individual (análisis unitario) o en conjunto (crítica comparativa). Por esto mismo, Bogost considera que el dominio de la retórica procedural no se reduce a los videojuegos: «I should stress that I intend the reader to see procedural rhetorics as a domain much broader than that of videogames, encompassing any medium -computational or not- that accomplishes its inscription via processes» (2007, p. 59). Sin embargo, representar procedualmente sí es característico de la computación, y aún más de los videojuegos (Bogost & Wright, 2007). Y es que los procesos, entendidos como operaciones unitarias, se ejecutan, es decir, son en tanto que funcionan. Los procesos *son* en acto. Así, según esta lectura, la principal diferencia entre los videojuegos y cualquier otro medio expresivo es que solo en el videojuego el proceso es en acto. En los demás casos, la representación nunca llega a ser procedural: en el texto se *describen*, en el cine se *muestran*, en la fotografía se *capturan*, en los videojuegos, en cambio,

se ejecutan. Esto explica por qué los videojuegos pueden llegar a ser simulaciones particularmente efectivas, pues los sistemas representados son en acto, de la misma manera que el sistema fuente. De este modo, si para la perspectiva digital, la especificidad de lo digital es la manipulación de la electricidad por medio del control de su intensidad, frecuencia y transformación (Maldonado & Rodríguez, 2017), para Bogost, la especificidad de los videojuegos sería su capacidad de representar procedualmente procesos con los cuales es posible interactuar en un espacio de posibilidades. Este espacio es construido por seres humanos, de tal manera que transmiten ideas, emociones y conceptos, lo cual constituye su expresividad. La retórica procedural es un intento por sacarle el máximo provecho a la capacidad representativa de los videojuegos por medio de la configuración minuciosa de su espacio de posibilidades para una expresividad más efectiva. En este sentido, con este tercer punto se introduce la expresividad como elemento propio de los videojuegos, o más precisamente, como una de sus funcionalidades, como se argumenta en la última sección.

En cuarto lugar, los videojuegos poseen la funcionalidad de ser expresivos en sí mismos. En la medida en que están constituidos por operaciones unitarias, los videojuegos ya poseen, por decirlo de alguna manera, una lógica interna. La crítica de Bogost (2008c; 2007) al movimiento de los juegos serios tiene su razón de ser en que al separar, conceptualmente, lo lúdico y lo serio, se asume que cualquier finalidad es externa al videojuego, de modo que, por ejemplo, un videojuego educativo no es educativo *per se*, sino que lo es porque el carácter educativo le fue añadido. Con afinidad a la lectura de la Ontología Serial del Software sobre las funcionalidades externas impuestas a lo digital, Bogost subraya la necesidad de que cualquier finalidad (o en los términos de este trabajo, funcionalidad), como la educativa, surja de la propia estructura y mecánicas del videojuego de manera orgánica. Se deben explorar las funcionalidades propias de

lo digital en general, y de los videojuegos en particular, sin acudir al uso de parámetros que no les corresponden, y que, peor aún, pueden malograr su comprensión y su uso.

Entonces, los videojuegos destacan por su capacidad de representar procesos con los cuales es posible interactuar en un espacio de posibilidades. Esto constituye la capacidad para ejecutar procesos que pueden transmitir mensajes, de manera que la expresividad es otra característica fundamental de los videojuegos. A partir de aquí es posible cuestionar cómo los videojuegos pueden ser usados, en nombre de su expresividad, para difundir ideas que no les corresponden, toda vez que se ignoran sus potenciales funcionalidades en tanto tecnología digital. Los videojuegos son, en efecto, expresivos, pero no de cualquier manera. Ha de revisarse, por un lado, su dimensión serial y estructural, las cuales conjugan elementos tanto simbólicos como materiales, y por el otro, sus capas de intencionalidad, dado su *telos* semi-abierto, hasta llegar a los múltiples *telos* en el nivel del usuario-jugador. Es necesario, en fin, apuntarle a una ontología del videojuego en tanto software, esto es, en tanto tecnología digital.

4. Conclusiones

Los videojuegos son una moda social y académica (Maldonado & Cáceres, 2023). Basta revisar, por un lado, las cifras producto de las ventas de videojuegos, las cuales no paran de incrementar con el paso de los años (Gracia Bonil, 2022) y, por el otro, la cantidad de producción académica que los toma como objeto de estudio. Sin embargo, su aproximación desde la filosofía no debe tomarse como el mero seguimiento de una tendencia. Los videojuegos son software, y el software es complejo; comprende, al menos, dos dimensiones y cuatro series. En este sentido, el estudio filosófico de los videojuegos es meritorio no porque se trate de una moda *per se*, sino porque se han introducido en la cotidianidad hasta el punto de convertirse en una moda de doble sentido, a saber, social y académica. El elemento de la moda, si bien está ahí, no es el factor decisivo que le otorga completo mérito al estudio filosófico de los videojuegos. Más bien, los videojuegos ameritan su abordaje filosófico porque se trata del fenómeno de lo digital por antonomasia. El presente trabajo se encaminó en esa dirección.

La investigación se llevó a cabo en tres capítulos. En el primer capítulo se hizo una exposición de la Ontología Serial del Software como método para el análisis. Se presentaron sus mayores aciertos, así como sus alcances para los propósitos del presente trabajo. En el segundo capítulo se presentó la teoría de Ian Bogost en relación con la pregunta ¿qué es un videojuego? Se trata de una sugerencia de lectura de Bogost mediante la estructuración particular de su teoría plasmada, especialmente, en los libros *Persuasive Games: the Expressive Power of Videogames* (Bogost & Wright, 2007), *Unit Operations: An Approach to Videogame Criticism* (Bogost, 2008c) y *Racing the Beam: the Atari Videogame Computer System* (Montfort & Bogost, 2009). Después,

en el tercer capítulo se llevó a cabo un análisis de la estructura teórica de Bogost, planteada en el segundo capítulo, mediante la Ontología Serial del Software presentada en el primer capítulo. Se presentaron sus similitudes, sus divergencias y cómo desde el lente de la Ontología Serial del Software se puede dar una interpretación más adecuada de algunos de los elementos de la teoría de Bogost. Se espera que este acercamiento haya logrado su cometido, esto es, aportar a los estudios sobre los videojuegos desde la lectura particular de un autor.

Por otro lado, como se sugirió en el primer capítulo, la labor investigativa sobre este autor no se agota con lo expuesto aquí. Tanto la Ontología Serial del Software como Ian Bogost continúan en estado propositivo. De hecho, se es consciente de que en este trabajo no se abordó en su totalidad el concepto de operación unitaria de Bogost. El filósofo se exploya en su interpretación del término en *Alien Phenomenology* (2012), en donde presenta su versión de una ontología orientada a objetos. De acuerdo con Bogost (2008a), esta propuesta es lo suficientemente distinta y compleja con respecto a su trabajo anterior como para ser asumida como un tercer método de análisis de videojuegos. En este enfoque se pone mayor énfasis en los objetos mismos y sus interacciones que en el ser humano, quien, dicho sea de paso, formaría parte de esta red de interacciones en la forma de un objeto más, sin mayor o menor importancia que el resto. Bogost argumenta que la idea de objetividad que se pretende en la investigación produce un distanciamiento, en lugar de un acercamiento, a la comprensión de la experiencia de una entidad, es decir, impide comprender cómo se da la experiencia (cómo ve, cómo siente, etc.) en una entidad no humana: «Counterintuitive though it may seem, the characterization of an experience through supposedly objective evidence and external mechanisms leads us farther from, not closer to, an understanding of the experience of an entity» (Bogost, 2012, p. 63). Esto se debe a que la idea de objetividad en realidad se fundamenta en el punto de vista humano, y lo utiliza como parámetro

para el estudio de todos los seres, sean humanos o no. El distanciamiento que provoca este proceder también ocurriría en los estudios sobre videojuegos. No obstante, Bogost asegura que con su análisis de plataformas ya se había encauzado a una ontología orientada a objetos contraria a los estudios fundamentados en la idea de objetividad (Bogost, 2008a; Williams, 2019). Más aún, el concepto de operación unitaria, el cual fue formulado en oposición al de operaciones del sistema en el método de análisis de la proceduralidad (Bogost, 2008c), también es una prueba de este encauce, dadas las propiedades de la discreción, la compresión y la conmutabilidad de la operación unitaria que allí plantea. Así, la *OOO* siempre ejerció influencia en la producción intelectual de Bogost, y quizás, como él mismo sugiere, de una combinación de ambos métodos de análisis previos se pueda formular el tercero: «Both remain human-centric affairs, ones concerned with the creation and reception of computer media by ordinary people. But, a combination of these two inches closer to a phenomenology of videogames.» (Bogost, 2008a, p. 32). Sin embargo, para los modestos propósitos de este trabajo, la revisión de este supuesto tercer método de análisis de videojuegos no resultó necesario. De hecho, en la medida en que se revisaron los dos primeros métodos de análisis en conjunto, se considera que, inadvertidamente y con sus propias particularidades distintivas, se ha seguido un camino similar al de Bogost. En este sentido, se estima que una revisión, en propiedad, de lo expuesto en *Alien Phenomenology* (2012) no menoscaba la validez de lo presentado en este trabajo. Antes bien, se trataría de una investigación complementaria.

Asimismo, lo que se ha dicho sobre las investigaciones complementarias a la teoría de Bogost también se aplica a la Ontología Serial del Software. De hecho, la idea de Bogost sobre una ontología orientada a objetos, es decir, una ontología que preste especial atención a los objetos mismos sin la intervención humana, ya forma parte de la teoría de la Ontología Serial del Software:

«Los objetos digitales son entes autónomos, son realidades autosuficientes que se independizan de su creador; de un modo relativo, claro está.» (Maldonado & Rodríguez, 2015, p. 39). Y es que, llegado un punto, los objetos funcionan por sí mismos, de una manera tal que, desde la perspectiva humana, sería completamente alienígena. No obstante, quedaría por revisar si desde Ontología Serial del Software, Bogost esté en lo correcto al proponer una *fenomenología*, lo cual parece una aproximación a los objetos tan humanamente sesgada como las que se pretenden evitar. Por otro lado, cabe recordar que la Ontología Serial del Software se sustenta en una ontología social, la cual, aunque brevemente explorada aquí cuando se revisó el fenómeno de la sobre-codificación humana, aún no se ha planteado formalmente y en su totalidad entre los mismos autores de la propuesta. Así, pues, estos dos elementos, a saber, la validez de una fenomenología y la conexión entre la Ontología Serial del Software y una ontología social, también quedan como parte de futuras investigaciones complementarias.

Referencias Bibliográficas

- Aarseth, E. J. (1997). *Cybertext: Perspectives on Ergodic Literature*. JHU Press.
- Abt, C. C. (1987). *Serious Games*. University Press of America.
- Acevedo-Merlano, A. A., & Chaux Lizarazo, J. A. (2016). Aproximaciones a los videojuegos y su incidencia en las subjetividades de los jugadores latinoamericanos. *Aposta: Revista de ciencias sociales*, 69, 140-157.
- Blakey, H. (2021). Designing Player Intent through “Playful” Interaction. *M/C Journal*, 24(4).
<https://doi.org/10.5204/mcj.2802>
- Bogost, I. (2008a). *The phenomenology of videogames*. 22-43. <https://publishup.uni-potsdam.de/opus4-ubp/frontdoor/index/index/docId/2553>
- Bogost, I. (2008b). The Rhetoric of Video Games. En K. Salen (Ed.), *The Ecology of Games: Connecting Youth, Games, and Learning* (pp. 117-139). MIT Press.
- Bogost, I. (2008c). *Unit operations: An approach to videogame criticism*. MIT Press.
- Bogost, I. (2012). *Alien phenomenology, or, What it's like to be a thing*. University of Minnesota Press.
- Bogost, I. (2015). *How to Talk about Videogames*. University of Minnesota Press.
<https://doi.org/10.5749/j.ctt184qqnb>
- Bogost, I., Ferrari, S., & Schweizer, B. (2010). *Newsgames: Journalism at Play*. MIT Press.
- Bogost, I., & Wright, W. (2007). *Persuasive Games: The Expressive Power of Videogames*. MIT Press.
- Burke, K. (1969). *A Rhetoric of Motives* (1ra ed.). University of California Press.

- Caillois, R. (1986). *Los juegos y los hombres. La máscara y el vértigo*. Fondo de Cultura Económica.
- Cardero, J. L. N., Rufi, J. P. P., & Pérez, F. J. G. (2014). El pensamiento abductivo como fundamento ontológico de los videojuegos. *Revista ICONO14 Revista científica de Comunicación y Tecnologías emergentes*, 12(2), 416-440. <https://doi.org/10.7195/ri14.v12i2.670>
- Collantes, X. R. (2013). Juegos y videojuegos. Formas de vivencias narrativas. En C. Scolari (Ed.), *Homo videoludens 2.0: De Pacman a la gamificación* (Nueva ed). Universitat de Barcelona, Laboratori de Mitjans Interactius.
- Corona, A. (2014). Bogost, I. (2006). Unit Operations. Cambridge, MA: The MIT Press. *Global Media Journal México*, 11(22), 79-80.
- CU Boulder Libraries (Director). (2011, noviembre 22). *Ian Bogost: Materials: Books, Games, Media and other Platforms—YouTube*. <https://youtu.be/v8IND06TosQ>
- Decuir, J. (1977). *Video Olympics* (Versión de Atari VCS) [Software]. Atari Inc.
- Edge of Reality. (2000). *The Sims* (Versión para Microsoft Windows) [Software]. Electronic Arts.
- Frasca, G. (1999). *Ludology Meets Narratology: Similitude and differences between (video)games and narrative*. <https://ludology.typepad.com/weblog/articles/ludology.htm>
- Frasca, G. (2003). *Ludologists Love Stories, too: Notes from a Debate that never Took Place*.
- Frasca, G. (2001). *Simulation 101: Simulation versus representation*. ludology. <https://ludology.typepad.com/weblog/articles/sim1/simulation101b.html>
- Gavarini, A. (2015). El pensamiento sobre la técnica de Gilbert Simondon. *Tecnología & Sociedad*, 4, 11-36.

- Gracia Bonil, S. L. D. (2022). *Data analytics: Sobre el comportamiento de la industria de videojuegos en los últimos cuarenta años*.
<http://repository.unimilitar.edu.co/handle/10654/44737>
- Grimes, S. M., & Feenberg, A. (2009). Rationalizing Play: A Critical Theory of Digital Gaming: Critical theory of communication technology. *The Information Society*, 25(2), 105-118.
- Hill, C. A. (Ed.). (2004). *Defining visual rhetorics*. Lawrence Erlbaum.
- Juul, J. (2011). *Half-Real: Video Games between Real Rules and Fictional Worlds*. MIT Press.
- Juul, J. (2016). Playing. En R. Guins & H. Lowood (Eds.), *Debugging Game History—A Critical Lexicon* (pp. 351-358). MIT Press.
<https://direct.mit.edu/books/book/3495/chapter/117681/Playing>
- Kulovitz, K. (2013). Cyberbullying in «Left 4 Dead 2»: A Study in Collaborative Play. *Theses and Dissertations*. <https://dc.uwm.edu/etd/363>
- La Hera, T., Jansz, J., Raessens, J., & Schouten, B. (2021). *Persuasive Gaming in Context*. Amsterdam University Press. <https://doi.org/10.5117/9789463728805>
- Maietti, M. (2013). Idea y vuelta al futuro. El tiempo, la duración y el ritmo en la textualidad interactiva. En C. Scolari (Ed.), *Homo videoludens 2.0: De Pacman a la gamificación* (Nueva ed). Universitat de Barcelona, Laboratori de Mitjans Interactius.
- Maldonado, J. (2020). Editorial. La filosofía frente a la tecnología computacional digital o la invención de la digitalidad. *Revista Filosofía UIS*, 19(1), 11-20.
<https://doi.org/10.18273/revfil.v19n1-2020014>
- Maldonado, J. (2019). *Desde una ontología serial del software: La cuarta serie de la agencia humana*. XVIII Congreso Interamericano de Filosofía, Universidad del Rosario.

- Maldonado, J., & Cáceres, P. (2023). Editorial. Videojuegos: Un reto para la filosofía. *Revista Filosofía UIS*, 22(2), 11-22. <https://doi.org/10.18273/revfil.v22n2-2023018>
- Maldonado, J., Rodríguez, D., Cáceres, P., & Suárez, J. (2020). An Ontology of Software: Series, Structure and Function. *Praxis Filosófica*, 51, 115-132. <https://doi.org/10.25100/pfilosofica.v0i51.10114>
- Maldonado, J., & Rodríguez, D. (2015). Humanidad y universo digital: Prolegómenos al problema ético de la utilidad y el perjuicio de lo digital para la vida. *Análisis*, 46(84), 27-40. <https://doi.org/10.15332/s0120-8454.2014.0084.02>
- Maldonado, J., & Rodríguez, D. (2017). Critical digitality: From the virtual to the digital. *Praxis Filosófica*, 45S, 145-163. <https://doi.org/10.25100/pfilosofica.v0i45S.6134>
- Montfort, N., & Bogost, I. (2009). *Racing the Beam: The Atari Video Computer System*. MIT Press. <https://doi.org/10.7551/mitpress/7588.001.0001>
- Murray, J. H. (2017). *Hamlet on the Holodeck, updated edition: The Future of Narrative in Cyberspace*. MIT Press.
- Nelson, M. (2012). *Sicart's «Against Proceduralism»*. A reply. https://www.kmjn.org/notes/sicart_against_proceduralism.html
- Penny, S. (1996). The Darwin machine: Artificial life and interactive art. *New Formations*, 1996(29). <https://simonpenny.net/1990Writings/darwinmachine.html>
- Petrowicz, M. (2014). Ludo-narratywizm, czyli proceduralizm Bogosta na tle sporu ludologii z narratologią. *Replay. The Polish Journal of Game Studies*, 01, 81-91.
- playableUCSC (Director). (2013, septiembre 24). *Ian Bogost—Procedural Rhetoric (Media Systems #7)—YouTube*. https://youtu.be/VFaqguc_uNk
- Searle, J. (1997). *La Construcción Social de la Realidad* (3ra ed.). Paidós.

- Searle, J. (2017). *Actos de habla: Ensayo de filosofía del lenguaje* (1ra ed.). Cátedra.
- Sicart, M. (2011). Against Procedurality. *Game Studies*, 11(3).
https://gamestudies.org/1103/articles/sicart_ap
- Simondon, G. (2008). *El Modo de Existencia de Los Objetos Técnicos* (1ra Edición). Prometeo Libros.
- Spinoza, B. (2014). *Ética Demostrada Según el Orden Geométrico* (1.^a ed.). Tecnos.
- Valve. (2009). *Left 4 Dead 2* (Versión de Steam) [Software]. Valve.
https://store.steampowered.com/app/550/Left_4_Dead_2/
- Wagner, L., & Whitehead, B. (1979). *Video Chess* (Versión de Atari VCS) [Software]. Atari Inc.
- Warren, R. (1979). *Adventure* (Versión de Atari VCS) [Software]. Atari Inc.
- Williams, J. J. (2019). Games, Things, and Theory: An Interview with Ian Bogost. *Symploke* (Bloomington, Ind.), 27(1-2), 469-485. <https://doi.org/10.5250/symploke.27.1-2.0469>