

PONTIFICIA UNIVERSIDAD JAVERIANA  
FACULTAD DE ARTES

(t)error  
por Camilo Aguilar

Bogotá  
2009

# Agradecimientos

---

Agradezco a:

Mi hermana Juliana Aguilar quién ha sido mi mano derecha en este proceso,

Mi padre y mi madre quienes me han dado un apoyo incondicional,

Veronica Cuello por su ayuda pero sobre todo por su paciencia,

Ricardo Toledo por su interés en mi proyecto y por su acertividad como asesor,

Jose Alejandro Lopez con quien empecé este proceso y me dio buenas herramientas para continuarlo,

Leonor Villaveces y a Hamilton Mestizo quienes me extendieron una mano en las temáticas tanto técnicas como conceptuales,

Pablo Camacho, Margarita Acosta, Julian Valdivieso, Andrea Barragan, Sandra López y todas las personas que participaron en mis experimentos, ya fuera de forma voluntaria o como 'victimas' de ellos.

# Índice

---

Agradecimientos.....	2
Introducción.....	4
Juego y jugador.....	9
Código.....	21
Eliza y el sentido.....	29
Parásitos.....	41
Imaginarios.....	53
(t)error.....	65
Bibliografía.....	77

# Introducción

---

El punto de partida de mi investigación fue un experimento que denominé *espacioabierto*, el cual consistió en hacer públicos el nombre de usuario y contraseña de mi cuenta dentro del (entonces) popular servicio de Internet *Hi5*. Este servicio, para describirlo someramente, le permite crear un perfil personal y relacionarlo con otros contactos ('amigos') pertenecientes a dicha red. Con el nombre de usuario y la contraseña, las personas tenían acceso a 'mi' cuenta y podían modificar lo que quisieran en ella: información personal, fotos, escritos y los demás elementos que un usuario puede manipular en este portal, como por ejemplo el envío de mensajes a mi nombre y el enlace con nuevos contactos.

El resultado de dicho experimento fue que los 'yo alternos' (las intervenciones hechas por otros) se establecían como coordinadas 'culturales' y/o ideológicas en las cuales, directa o indirectamente, yo estaba involucrado. Lo primero que se hizo fue cambiar mi preferencia sexual (de hetero a homo) y paso seguido, mi perfil dejó de ser el de Camilo Aguilar para convertirse en el de Amparo Grisales. Al collage se agregó información personal de otras personas, relaciones con Al Qadea y hasta resulté ser fan de Daniela la protagonista de *Padres e Hijos*.

En cuanto a metodología, este experimento consistió en remplazar a un actor (yo) por otros actores pertenecientes a un contexto cultural cercano (los otros usuarios registrados que intervinieron el perfil), lo cual implicó un abandono a aquella *meta-identidad* de los sistema representacionales que abundan en Internet. Este intercambio de roles y la tensión entre abandono y apropiación marcaron la formula metodológica que me guió a plantear nuevos ejercicios y experimentos.

Por ejemplo, en otro ejercicio llamado *No soy yo, eres tú* (el cual mencionaré en otros capítulos de este texto), permití que un ente de 'inteligencia artificial' (estupidez virtual llamado *Eliza*) se encargara de mis conversaciones dentro de un sistema de mensajería instantánea por Internet. En este caso, se remplazó a un actor humano por un actor máquina, lo que trajo consigo un nuevo elemento: si bien en

*espacioabierto* la máquina funcionaba cómo medio y establecía el contexto, en *No soy yo, eres tú*, la máquina era un ente activo dentro de la relación con los actores humanos.

Sin embargo, la máquina como actor no es algo que sólo se dé dentro de mi ejercicio: es recurrente ver como las máquinas y los sistemas son estructurados para que lleven acabo actividades que podrían también ser efectuadas por humanos (en muchos casos hasta remplazan al humano en estas tareas). Un caso que lo ejemplifica son las funciones de corrección de ortografía pertenecientes a los procesadores de texto: si bien la corrección podría ser hecha por un humano, este le permite a la máquina que lo haga por él.

Por lo tanto, los programas de computador y las máquinas en general, no son sólo medios o contextos donde se posibilitan ciertas acciones humanas, sino que también se comportan como un ente que lleva acabo funciones que afectan las relaciones que se establecen a través de ellos.

Otro caso donde sucede esto es los sistemas ‘colaborativos’ que se encuentran en Internet, donde (se supone) el conocimiento es ‘elaborado por todos y todas’. Una de las críticas que se le hacen a estos sistemas (por lo menos a la mayoría) es que imponen de un sistema de consenso que no se lleva a cabo por medio del acuerdo social (como normalmente se hace) sino que son los códigos los que se encargan de categorizar los ‘conocimientos’ y conductas dentro de la ‘comunidad’ que posibilitan. Acá vemos cómo el código se comporta tanto como actor como contexto.

Teniendo en cuenta a estos sistemas mi trabajo se desligó de la idea de un ‘yo’ propia de los sistemas de auto-identificación, y se centro en la idea de un ‘nosotros’. A partir de la crítica mencionada, realice un ejercicio parodia de Wikipedia donde un programa tomaba frases escritas por el usuario, las descomponía en palabras y buscaba una por una en Wiktionary en español (plataforma hermana de Wikipedia que funciona como un diccionario) dando como resultado un significado críptico de dicha frase. También desarrollé un programa de computador que buscaba en Google las frases “we hate the” y “we love the”<sup>1</sup>, tomaba los resultados (“we hate the usa”,

“we love the israeli”, “we hate the 80s”, ...) y por medio de una voz robótica los enunciaba, indicando una atmósfera divergente en un medio consensual.

Los ejercicios que he comentado son los que denominaría como la primera etapa de mi investigación. Aunque creo que tenían cierto potencial y que la idea era clara, siempre se sentía que algo más estaba emergiendo de ellos: todos enunciaban una problemática de sentido. El collage en *espacioabierto*, la incongruencia del ente de inteligencia artificial, el críptisimo de las definiciones en el ejercicio de Wiktionary y la contradicción en los enunciados robóticos provenientes de Google.

Revisando mis ejercicios sentía que emergía un tema que se relacionaba con los que estaba tratando, pero que se ubicaba en un nivel más abajo. Entonces, mi investigación se disipó de las que habían sido mis intuiciones iniciales y se centró en una pregunta por la relación de sentido que establece un humano con un computador, pero sobre todo en una relación de sentido en falla, en error – en la incoherencia de los resultados de mis ejercicios.

En este texto expongo el proceso de investigación que abordé desde esta perspectiva, el cual me ha llevado a considerar el error en la máquina como un (t)error en la relación de sentido que establece con una persona.

En el primer capítulo, planteo la diferencia entre las relaciones humano-humano y las relaciones usuario-computador bajo el contexto del juego, para mostrar que en la relación con la máquina los motivos de actuación son diferentes: mientras que el humano tiene la posibilidad de actuar por y por medio del *sentido* propio, el programa de computador lo hace porque hay una serie de reglas que dictan su comportamiento. Luego, para comprender qué permite que la máquina ‘juegue’ (ya que no es el *sentido*) con el humano, en la segunda parte, se muestra qué entendemos por programa de computador y cómo este funciona tanto como contexto como ‘jugador’ (*agente*), pero para que la interacción sea posible es necesario que el humano interprete las acciones de la máquina. En el capítulo tres, se profundiza sobre esto intentando definir dónde está el sentido en la relación y se concluye que este recae sobre el usuario. En el cuarto se contrastan tres casos de parásitos comenzando desde un parásito biológico que actuó por medio de la afectación (no

funciona por medio del sentido) para expandirlo a otros casos donde la línea ente afectación por sentido y afectación mecánica se desdibuja y confunde.

A partir de los elementos anteriores, en el quinto capítulo planteo la relación usuario-computador, cuál es la génesis del usuario en el computador, los alcances de comprensión del funcionamiento de la máquina por parte del humano y qué pasa en la interacción cuando se obtienen respuestas inesperadas por parte de la máquina. Por último, expongo mi trabajo final y su desarrollo como consecuencia de los diferentes acercamientos al problema en la relación humano-máquina.





# Juego y jugador<sup>\*</sup>

---

Yo solía jugar con cierta frecuencia ajedrez pero desde que dejé de practicarlo he perdido ridículamente las pocas partidas que he iniciado. Por esta razón decidí que no volvería a jugar contra otra persona hasta que no mejor mis habilidades en el juego. Para practicar decidí descargar un juego de ajedrez para computador con el cual, siendo mi contrincante una máquina, podía afinar mi concentración y habilidades, ahorrándome la pena de cometer errores tontos delante de otra persona.

Busqué versiones sencillas (y gratuitas) de dicho juego: nada de gráficas ostentosas ni de interfaces sofisticadas, sólo lo necesario para identificar las piezas y el tablero, sin embargo, sí buscaba un buen jugador. Al principio pensé que no sería difícil ya que los programas de computador tienen fama de ser los mejores, pero, para mi sorpresa y decepción, la primera versión con la que me topé resultó ser un pésimo jugador: a forma de prueba, apliqué la jugada que se conoce como 'mate de pastor'<sup>1</sup> con la cual le gané. Decidí desinstalar este programa y buscar otro.

El segundo, cumplió con todas mis expectativas: nunca le he podido ganar (en algo tendrá que ver mi nivel de juego y concentración, pero por lo menos no le gano con el 'mate de pastor') y tuvo tanto éxito conmigo que, por un largo tiempo, se convirtió en mi hobby: jugaba entre 5 y 20 partidas al día.

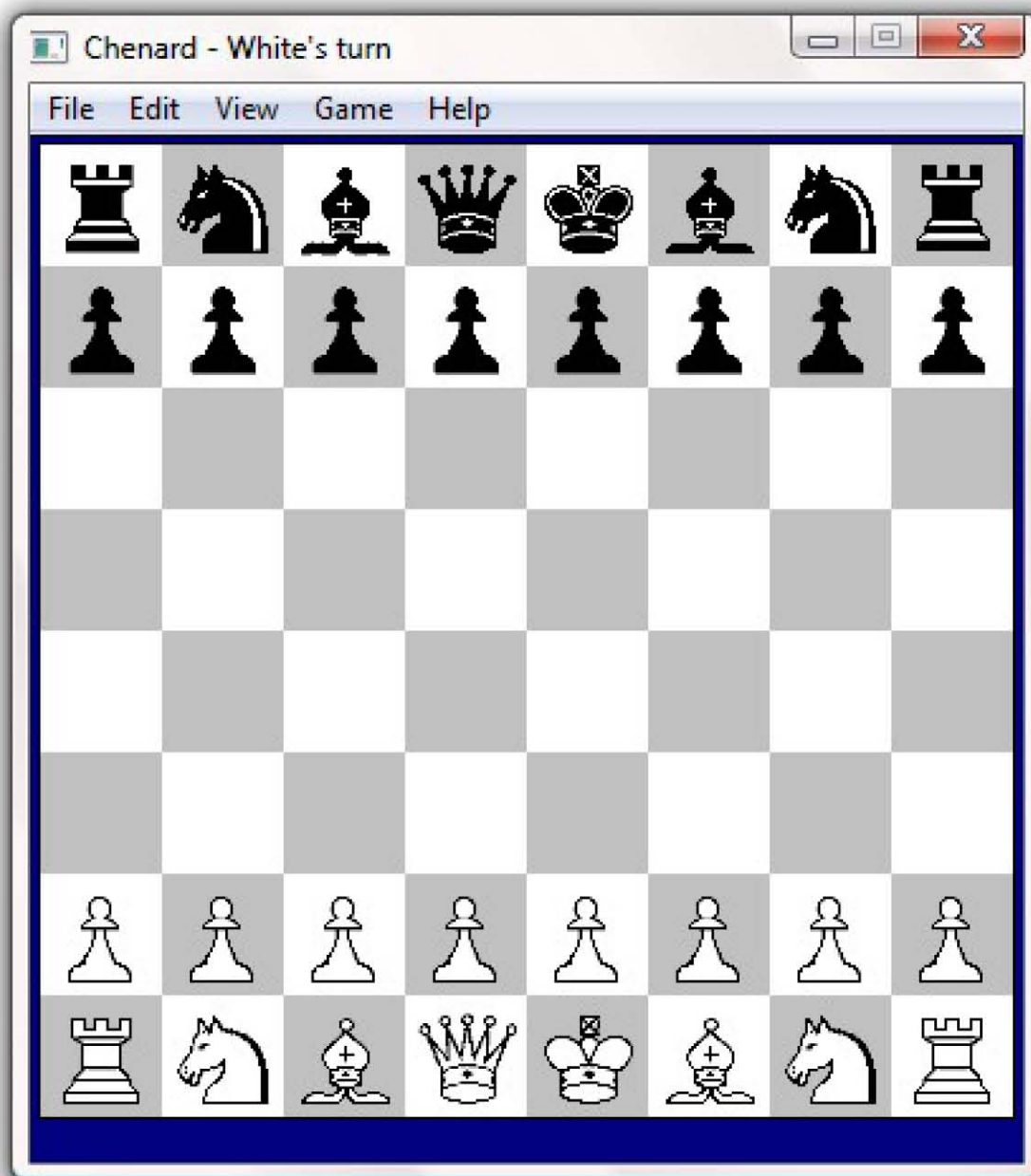
Gracias a la práctica constante mis habilidades para el juego mejoraron (tal cual lo esperaba) . Si en un principio perdía a los pocos movimientos, en unas semanas ya empezaba a jugar de forma estratégica y previsiva, hasta llegué a comer todas las piezas 'grandes' de mi contrincante sin cuerpo, aunque después coronó dos peones y me ganó.

Sentía que ya estaba preparado para probar mis habilidades contra otra persona, así que reté a una amiga que en ese momento tenía 'fiebre de ajedrez'. Pensé que, si no podía darle la talla, por lo menos no iba a cometer errores tontos.

---

\* Este capítulo está basado en las INVESTIGACIONES FILOSÓFICAS de Ludwig Wittgenstein y de la introducción del libro LECCIONES Y CONVERSACIONES SOBRE ESTÉTICA, PSICOLOGÍA Y CREENCIA RELIGIOSA donde Isidoro Reguera comenta los puntos centrales de la filosofía de este autor.

1 En ajedrez el 'mate de pastor' es una jugada que permite ganar en pocos movimientos, pero que es sumamente obvia y caen en ella sólo algunos principiantes desprevenidos



*Chenard. Juego de ajedrez para computador.*

Sin embargo, una vez más, jugué al mejor estilo suicida - regalar la reina al poco tiempo de empezar la partida ya era augurio de que me vencería sin mayores complicaciones. Una vez más había hecho el ridículo jugando ajedrez.

Este resultado indica que mi entrenamiento no fue muy efectivo. Pensaba que (por lo menos) no iba a 'regalar el juego', pero jugué de la misma forma que lo había hecho antes: mal. A pesar de que no hay una razón clara (o excusa si se quiere) por la cual esto sucedió hay dos puntos interesantes a la hora de comparar ambas circunstancias (contra un programa y contra un humano):

### **1. El juego**

Por un lado, es curiosa como se emula el ajedrez de mesa en la pantalla del computador. Lo primero que notamos es que las piezas son planas y se entienden más como esquemas (son iconos) que como objetos. Sin embargo esto no es una diferencia crucial con el juego de mesa, ya que en un juego de ajedrez no importa si las piezas están hechas de madera, marfil o si tienen tal o cual forma, como tampoco importa si son objetos o representaciones iconográficas, ya sean dibujadas en trozos de papel o correspondientes a un proceso digital. Lo importante es que las piezas sigan siendo piezas: que se puedan identificar y diferenciar unas de otras.

Pero, al igual que las piezas, varios de los elementos, que constituyen lo que yo llamo la *experiencia concreta del juego*, se presentan como esquemas. Con este término no sólo me refiero a las *reglas* que lo componen, sino también a ciertas prácticas que son determinadas por costumbre y/o practicidad para los jugadores. Una de ellas es disponer las piezas con las que se va a jugar (negras o blancas) en el lado donde se ubica el jugador. Podríamos pasar por alto esta costumbre (poner las piezas en el lado contrario del jugador) sin alterar el juego, pero debemos tener presente que esta disposición es la que estructura nuestro *entendimiento espacial* del juego. Por ejemplo, si le estamos enseñando a alguien a jugar ajedrez y (después de enseñarle los nombres de las piezas y su ubicación inicial en el tablero) queremos enseñarle como se mueve el peón, es muy probable que le indiquemos que éste se mueve hacia *adelante*, siendo este desplazamiento relativo a la ubicación de nuestro discípulo. Esto es reforzado por la *perspectiva* generada por el punto de vista que

tiene el jugador, ya que es la que establece el *allá* y el *acá* a los que se hacemos referencia al decir, por ejemplo, que el peón *avanza* (se aleja de nosotros).

Para emular este *entendimiento espacial* el programa utiliza varios elementos. Por ejemplo, hace uso de una franja azul para indicarle al jugador en qué lado está jugando. Pero lo que realmente fortalece la emulación espacial es que las piezas del jugador se disponen en la parte de abajo del tablero. Esto, por un lado, emula la forma como se vería el tablero si el jugador se parara de su asiento y lo observara de forma cenital, lo que mantiene la *percepción de profundidad*, y establece el 'allá' y el 'acá'. Por otro lado, hace que los movimientos de la mano sean más 'naturales' (son más parecidos a la convención) ya que el movimiento que se hace con el ratón mantiene las mismas coordenadas que el que hace con la mano sobre un tablero (compare como movería usted un peón en un tablero físico y como lo haría en el programa con un ratón).

Este tipo de emulaciones corresponden a la esquematización de la *experiencia del juego*, sin embargo esto no implica que el juego haya *cambiado*. En el ajedrez este tipo de diferencias son tan relevantes como la que hay entre jugar en una playa, un bar, una casa o un coliseo. Son diferencias que no mutan ni la forma de jugarlo ni sus reglas.

Con otro tipos de juegos, como el fútbol, la situación es diferente. En los video juegos de fútbol, a pesar de que existen las mismas reglas que en el juego físico (faltas, goles, tiros de esquina, amarillas, rojas, etc.), la experiencia cambia de tal forma que se hace evidente que no se trata del *mismo* juego. Jugar fútbol en un campo físico y jugarlo en una consola son definitivamente dos *juegos diferentes*. Es más preciso decir que el video juego se basa en el deporte (ni siquiera se le podría llamar deporte al video juego) y que por esta razón mantienen cierta *similitud*.

¿Por qué el ajedrez no muta mientras que el fútbol sí lo hace? El ajedrez se puede llevar a casos extremos como es el juego 'a ciegas' donde no se utiliza ningún elemento para mantener el estado del juego a excepción de la memoria de los jugadores (personas fuera del juego - árbitros - mantienen un registro de las jugadas en un tablero, pero los jugadores no lo ven) y aún seguir siendo ajedrez. En el fútbol

esto sería imposible ya que este requiere de elementos que constituyen su contexto pero que no hacen parte de sus reglas: la gravedad, la altura, el sol, la luz y hasta el agotamiento físico de los jugadores. Los elementos que hacen parte del contexto (no de las reglas del juego) no necesitan ser acordados o asumidos por sus jugadores. Imagínese a dos equipos de fútbol acordando que el balón debe caer, esta sería una situación ridícula ya que no hay forma de que se haga un acuerdo de lo contrario (a menos que el juego se llevara a cabo en el espacio o un planeta con poca gravedad, en cuyo caso tampoco se podría acordar que el balón cae).

En el juego de ajedrez por el contrario la única condición necesaria (lo único que se requiere del contexto) es que hayan dos entes en capacidad de competir, el resto del juego son reglas – acuerdos entre los jugadores. Si por ejemplo un jugador cayera de su silla y provocara que el tablero también cayera, esto sería algo fuera del juego. Indicaría su fin sin duda, pero sin embargo no si hay una ganador, un perdedor o un empate.

## **2. El jugador**

Ahora, yo decidí *practicar con* un programa de computador antes de *enfrentarme contra* una persona, lo cual indica que debe haber alguna diferencia entre las dos circunstancias, aun cuando el juego no cambie. Hasta en esta última frase hay algo que indica que hay diferencia: lo llamo practicar y no, por ejemplo, competir.

Y esta diferencia era clara desde el momento mismo en que tomé la decisión de jugar contra un programa de computador: lo que buscaba era un programa de ajedrez que, si bien no era sofisticado en cuanto a interfaz y gráficas, debía ser un contendiente de buen nivel – debía ser un buen ‘jugador’.

Pero debemos de advertir que esta diferencia no se establece en *cómo* el programa juega (a mencioné que el primer programa que descargué jugaba mal y que el segundo jugaba bien, por ejemplo). Para descartar esto podríamos asumir que el programa juega tal cual como lo hace una persona y además que aprende nuevas jugadas y estrategias con cada partida, pero aún así el juego se seguiría mostrando

diferente. Es más, una diferencia en el *como* sería volver a intentar encontrar el cambio dentro de las diferencias de *experiencia*.

Lo que se debe tener en cuenta es una característica *sui genesis* (por así decirlo) del contrincante digital, que se plantea desde la relación que establece con el jugador humano (que es diferente a la que éste último establece con un igual). En mi caso concreto la diferencia era que no existía responsabilidad o compromiso con el contrincante. Abandoné varias partidas (porque me aburría o porque tenía algo más que hacer) sin que esto repercutiera en lo que el contrincante pudiera esperar de mí. También variaba mi nivel de compromiso (algunas veces jugaba de forma concentrada y analítica mientras que otras lo hacía casi que al azar) sin que esto pudiera ‘aburrir’ o ‘defraudar’ al programa. En ningún momento se vio afectada la forma en que jugaba el programa por la forma en que yo lo hacía: siempre se tomaba el mismo tiempo analizando el estado del juego y respondía con una maniobra de la misma calidad.

Por el contrario, cuando jugué con mi amiga, consideraciones como “no puedo hacer jugadas tontas”, o el hecho de sentir que me estaba demorando mucho en una jugada, ejercían cierta presión a la cual no me enfrentaba cuando jugaba en mi computador. Las consideraciones hacia el contrincante humano empezaban a tener un peso que no existía a la hora de jugar contra una máquina. También supongo que mi mal juego repercutió en la forma de jugar de mi amiga: al ver que yo hacía movimientos estúpidos seguramente ella jugó de forma confiada y sin esforzarse mucho. También es posible que si volviera a pedirle que jugáramos, ella diría que no, ya que consideraría que no es un reto jugar contra mí.

En este caso, cuando el jugador hace parte del programa, ya no hablamos del mismo juego sino de una serie de derivaciones: entrenamiento, práctica, competencia con uno mismo, aprendizaje, investigación del funcionamiento, etc.

Sin embargo existe otro caso en el que esta división se vuelve borrosa: las partidas “humano contra máquina” (*Man vs. Machine*). Estos son eventos donde un programa sofisticado de computador se enfrenta a un jugador de ajedrez de alto nivel. Uno de los eventos que más se ha reseñado (y al que haré referencia en este



*Vista general del juego. Se aprecia la pantalla que muestra el estado del juego junto a los jugadores (Krámnick, Feist y Deep Fritz).*



*Krámnick jugando contra Deep Fritz (computador).*

texto) es el del duelo entre Vladimir Krámnick (jugador super-estrella) contra Deep Fritz (poderoso programa de ajedrez), llevado a cabo en el 2006 (anteriormente estos dos 'competidores' habían jugado en el 2002). En estas partidas el objetivo del juego es diferente al mío (entrenarme). Por ejemplo, a Krámnick se le ofreció un millón de dolares si ganaba y la mitad si perdía, también tenía la presión de mantener su estatus de buen jugador, como también el compromiso de dar una buena partida. Al igual que en las partidas persona contra persona (y a diferencia de mi caso), el objetivo de este juego era ganar.

Esto es reforzado cuando vemos que en estas partidas 'humano contra máquina' se mantienen varios elementos y convenciones propios de un juego normal de campeonato. Por ejemplo, se suele utilizar un tablero físico a pesar de que el programa tiene una interfaz sobre la cual se puede jugar. En dicho tablero el jugador humano realiza sus jugadas y el *operador* del programa (que en este caso fue Mathias Feist, uno de los creadores del programa) mueve las piezas que el programa le indica en pantalla (y viceversa).

También encontramos que, al igual que en la mayoría de las competencias profesionales de ajedrez, los espectadores tienen dos formas de visualizar el juego. La primera es la de los dos jugadores sentados uno frente al otro con el tablero en medio de ellos. La otra muestra el estado del juego de una forma similar a la utilizada en las interfaces de los juegos de ajedrez para computador (descritas anteriormente).

Estas dos visualizaciones ofrecen dos tipos de información diferente: (1) la proyección muestra el estado del juego en un tiempo discreto utilizando elementos esquemáticos, y (2) la imagen de los jugadores sentados frente al tablero suscita una experiencia continua en la que se aprecia, no la calidad de las jugadas (si estas son buenas o malas), sino elementos con los que el espectador puede relacionarse directamente (expresiones, la forma en que se mueve, el nivel de concentración, etc.). Estos elementos no tienen ningún efecto concreto en el juego (no hacen que una jugada sea mejor o peor) y sin embargo son los encargados de producir una atmósfera de *espectáculo* (ver un juego de ajedrez no debe ser muy digerible).



El computador no está en capacidad de producir este tipo de espectáculo como tampoco el programa de suscitar dicha experiencia de continuidad. No estoy seguro de que esta sea la razón por la cual se mantienen estas convenciones cuando no son necesarias (se puede jugar directamente sobre la interfaz del programa). Sin embargo el hecho de que el operador sea uno de los creadores del programa implica que, de cierta forma, se ha logrado conmensurar el juego (contra máquina) con la experiencia de un juego de torneo, ya que el operador está en capacidad de exhibir un comportamiento continuo con el cual los espectadores (y el contrincante) se pueden relacionar.

Pero a pesar del grado de similitud al cual se puede llevar este tipo de partidas, seguimos pensando (de una manera intuitiva o no) que cuando Krámnick se enfrenta a otro jugador de ajedrez humano, es una situación diferente a cuando se enfrenta a Deep Fritz. En un campeonato, por ejemplo, un programa de computador no podría entrar a los escalafones ni ser considerado “el mejor jugador del mundo”. Lo más probable es que ni siquiera podría ingresar al torneo, no sólo porque esto implique una ventaja o desventaja frente a los otros jugadores, sino por que el concepto ‘competir’ sería diferente al que usa cuando se enfrentan dos personas. Si por ejemplo un programa de computador entrara a un torneo de ajedrez, sus creadores serían los que competirían, ya que son los que estarían en capacidad de inscribirse, de tener presencia en el torneo y de recibir un premio. En otras palabras, serían ellos (al igual que Krámnick) los que tendrían motivaciones, razones e iniciativa, mientras que el programa simplemente se ejecutaría (lo ejecutarían) a su debido momento y actuaría, no por razones y motivos, sino porque fue diseñado y programado para hacerlo.

Esta diferencia es la misma que encontramos entre los tipos de respuesta posibles cuando se hacen estas dos preguntas: “¿por qué Krámnick juega?” y “¿por qué Deep Fritz juega?”. En el caso de Krámnick, lo más probable es que la respuesta sea acerca de las *razones* que él tiene para jugar (“por que le gusta mucho”, “por que le pagan bien” o un discurso extenso respecto a su amor por el juego). En el caso de

Deep Fritz, la respuesta sería algo similar a “juega porque fue diseñado y programado para ello”. Si a esto respondiéramos con la pregunta “¿y por qué fue diseñado y programado Deep Fritz para jugar ajedrez?” lo más seguro es que las repuestas sean referentes a las *razones* o motivos de sus creadores, sin que nos indiquen razones o motivaciones propias del programa. Esta es la brecha que se origina cuando no preguntamos por un *cómo* sino por un *porqué*.

Para no entrar en los detalles de un juego tan complejo como el ajedrez tomemos otro ejemplo que nos ayuda a ilustrar esto: el juguete que genera secuencias aleatorias de colores las cuales deben ser repetidas por el jugador. Este juguete está equipado con un sistema para generar las secuencias y otro para evaluar si el jugador las repitió correctamente. Acá el juguete, al igual que el programa de ajedrez, simplemente está reaccionando a comandos (encender, apagar, generar, evaluar), de los cuales no puede dar *razón* y sólo se pueden entender por medio de *causas*.

Una repuesta al *porqué* de este juego tendría que relacionarse con su mecanismo y la forma en que reacciona a los comandos que se le dan. Una respuesta posible a la pregunta “por qué genera secuencias aleatorias” sería “porque ha sido encendido”, lo cual podríamos verificar apagándolo y encendiéndolo de forma repetitiva. Si por el contrario, le preguntáramos al jugador “por qué repite las secuencias que el juguete genera” la repuesta podría ser “porque así se juega”, “porque me ayuda a desarrollar mis habilidades psicomotrices”, o cualquier otra, sin que haya la necesidad de hacer referencia su propio mecanismo (ejemplo: cómo funciona el cerebro).

Podría presentarse el caso en que las instrucciones del juego indiquen algo diferente a lo que el jugador está haciendo o que este juego no ayude a desarrollar habilidades psicomotrices. Consideraríamos que el contenido de la respuesta es falso, sin embargo esto no deslegitima la respuesta que dio el jugador, ya que lo que estaba en juego en esta pregunta era la razón por la cual él repetía las secuencias, y no la frase subordinada que lo acompaña<sup>2</sup>. Por lo tanto cuando se trata de dar *razón*

---

2 Otro ejemplo en el lenguaje: en la frase “le gusta el pastel porque es dulce” el hecho de que sea dulce o salado es una afirmación que no se pone en juego, ya que lo que se intenta establecer es por qué le gusta el pastel.

de un acto, no estamos hablando de algo que se pueda evaluar bajo el criterio 'verdadero o falso'. ¿Qué pasaría si la respuesta hubiera sido "por que es divertido"? Aún si este juego nos aburriera, no podríamos decir que la *razón* que el jugador dio sea falsa y que este juego no lo entretiene a él.

Sin embargo, sí podríamos argüir que la *razón* que dio no nos *convence* (no nos satisface). Si por ejemplo la razón que nos dio fue "porque es divertido", y vemos que el jugador muestra ciertas características de aburrimiento y además se distrae fácilmente, diríamos "pues no parece que lo divierta mucho". Si vemos al personaje en la misma situación y nos dice "porque así se juega" esta sería una *razón* más convincente ya que no tendríamos objeciones contra ella.

En este sentido el objetivo de las *razones* es convencer y por lo tanto generar aceptación. Estamos hablando de "un tipo de razonamiento más retórico [...] (convinciente) que lógico (dogmático): se trata de persuadir, de convencer a la gente, o incluso a uno mismo, del valor explicativo de algo y no [...] de creer que hemos descubierto algo identificable y concreto en alguna parte indetectable y concreta [...]” (REGUERA (COMENTANDO A WITTGENSTEIN), 1992)

Esta característica de las *razones* re-ubica el *sentido* de los conceptos al sacarlo de su posición trascendental (propia de cuando se intenta establecer una verdad) y lo ubica en el *uso* que se hace de ellos.

Ahora, un programa como Deep Fritz puede jugar ajedrez, pero no puede darnos ni comprender razones: para responder a un *porqué* de su forma de actuar tenemos que hacer diagnósticos y pruebas que nos revelen las *causas* de sus acciones (por que hizo esa jugada), de la misma forma en que un doctor determinaría el *porqué* de un dolor de estomago.



La incapacidad de *sentido* es el abismo que se interpone entre un humano y una máquina. El humano, aunque también esté sujeto a causas como la sinapsis nerviosa, tiene la posibilidad de actuar por y por medio del *sentido* propio (dar y recibir razones), mientras que el programa de computador lo hace porque hay una serie de reglas (código) que dictan su comportamiento.



# Código

---

Los programas de computador ‘juegan’ porque hay una serie de reglas que hacen que jueguen, no por que hayan decidido hacerlo. Estas reglas son el *código*<sup>3</sup> del programa.

Para analizar la ‘naturaleza’ del *código* me voy a remitir al caso de un pequeño ‘hack’ utilizado en diferentes versiones de Windows (95, 98 y xp) para remplazar la palabra ‘Inicio’ en el botón . Con este truco se puede hacer que este botón se muestre así , por ejemplo.

Este ‘hack’ consiste en modificar un archivo llamado *explorer.exe*, donde se encuentra el *código* responsable de que aparezca dicha palabra en este botón. Se le considera un ‘hack’ porque modificar este archivo implica un procedimiento inusual<sup>4</sup>, ya que su fin es ser ejecutado (interpretado por un computador) y no ser leído (o modificado) por el usuario.

Lo que se encuentra cuando se abre este archivo para su lectura, es una serie de caracteres que resulta incomprensible, a menos que se esté familiarizado con este tipo de notación. Sin embargo, también se encuentran algunos textos utilizados en la interfaz que el usuario opera (Windows), entre los que se encuentra la palabra ‘Inicio’.

El ‘hack’ consiste en borrar esta palabra y en su lugar escribir otra, teniendo en cuenta que la palabra debe ser remplazada por otra de igual longitud (ej. ‘Nestor’, ‘Fluido’, ‘Axioma’, ‘Camilo’ o ‘Avatar’ tienen seis caracteres al igual que ‘Inicio’, por lo

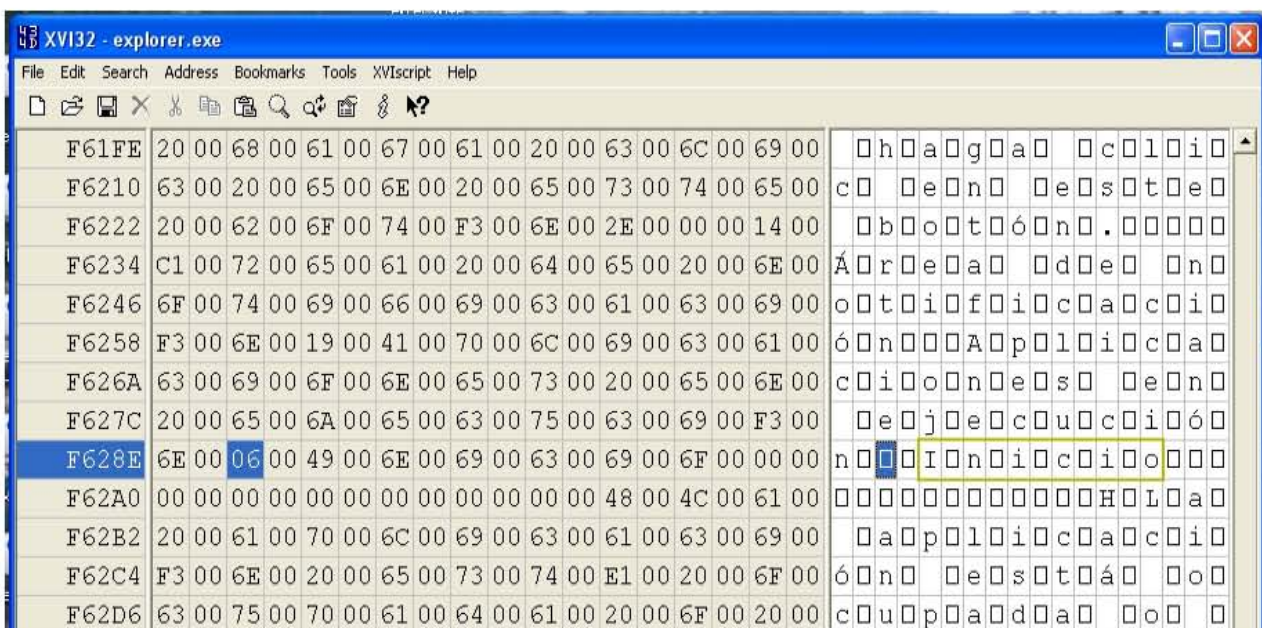
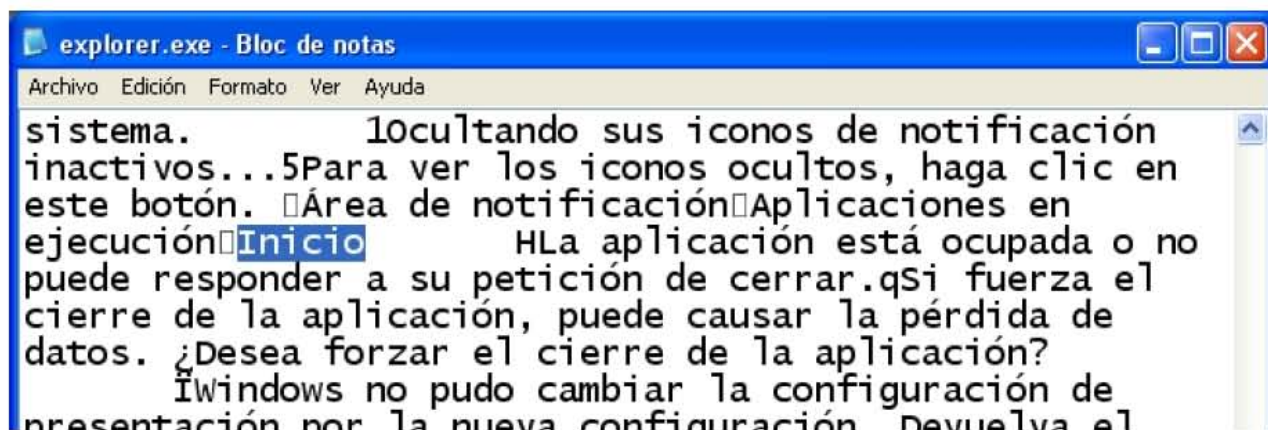
---

3 Al hablar de programas de computador hay básicamente dos tipos de código:

- *código fuente* - están escritos en un lenguaje simbólico (lenguaje de programación) utilizado para desarrollar los programas, pero que no pueden ser ‘ejecutados’ por el computador.
- *código compilado* – componen a los archivos ejecutables. Se generan al compilar (procesar, codificar) el código fuente.

En este texto, a menos de que se especifique, al utilizar la palabra código se hace referencia a códigos compilados.

4 Por lo general, este tipo de archivos no se modifican. Normalmente, para realizar algún cambio, se recurre a otros métodos. En este caso, la palabra ‘Inicio’ no puede ser modificada por ningún otro método a menos que se tengan los archivos que son usados para desarrollar los programas (código fuente) que son diferentes a los que se distribuyen (código compilado) (un usuario cualquiera de Windows no los tiene este tipo de archivos). Por esta razón se considera a este truco un ‘hack’ ya que implica que se está haciendo algo a través de un método no convencional.



Archivo explorer.exe siendo editado en:  
 Superior: bloc de notas (editor de texto),  
 Inferior: XVI32 (editor hexadecimal):

(<http://www.chmaas.handshake.de/delphi/freeware/xvi32/xvi32.htm> – una herramienta más fácil de utilizar es Resource Hacker - <http://www.angusj.com/resourcehacker/> ).

En la columna derecha aparecen los caracteres en formato ASCII y en la izquierda su valor hexadecimal. El carácter señalado en azul es el 'carácter prefijo' y delineado en verde se encuentra la palabra 'Inicio' (las letras tienen un valor vacío de por medio).

tanto son palabras válidas), de lo contrario esto podría ocasionar fallas en el sistema. En el caso de que quiera utilizar una palabra de longitud diferente, también se debe modificar el carácter ilegible que la antecede (en este texto lo llamaré ‘carácter prefijo’). Este es un número hexadecimal que especifica el largo de la cadena de caracteres que se quiere insertar (ej. 04 para ‘Pepe’, 0F para ‘Impenetrabilidad’, 00 si se quiere que no aparezca ninguna palabra).


Este ‘carácter prefijo’ es la *regla* dentro del *código* que determina la longitud que debe tener la palabra que se encuentra dentro del botón, así que al modificarlo afectamos dicha *regla*. Al igual que este carácter, la mayoría de los otros caracteres que componen este archivo (ya sean solos o en grupos) son *reglas* que rigen el funcionamiento del programa<sup>5</sup>.

Así que cuando hablamos de *códigos de programas* hay una similitud a cuando hablamos acerca de otros tipos de código como los penales, los de conducta, los éticos, los simbólicos, etc. Si, por ejemplo, comparamos un *código penal* con el *código de un programa*, encontramos que en ambos casos la enunciación de las reglas *instauran* lo que debe efectuarse bajo ciertos contextos y ante ciertos eventos. Sin embargo, en un *código penal* existe la posibilidad de que la acción a la que hace referencia la *regla* no se lleve a cabo, se aplique mal o un sujeto la incumpla sin ser procesado; cosa que no puede suceder con lo que se especifica en los *códigos de programas*, ya que estos son deterministas, es decir: en el *penal* se decreta lo que que se debe *hacer* y en el de *programas* se define lo que debe *suced*er.


Podríamos incluir a las *leyes físicas* en esta comparación ya que estas, al igual que los *códigos de programas*, rigen la forma en que los eventos ocurren. Sin embargo, las leyes físicas son *descubiertas* mientras que las reglas de un código de un programa son *impuestas* (y las de un *código penal* son acordadas). Esto implica que la proposición que enuncia una *ley física* puede ser falsa o verdadera sin afectar

---

<sup>5</sup> Los caracteres que no son ‘reglas’ (caracteres operacionales) son, al igual que la palabra ‘Inicio’, ‘recursos’. Estos son datos ‘hard coded’ (fuertemente programados) que pueden ser textos, imágenes entre otros, que el programa manipula o muestra. Difiere de otro tipos de datos donde que se encuentran en un archivo o base de datos independiente al programa y que se llaman a través de ‘referencias’.

en lo absoluto a su referente (podríamos enunciar que la Tierra es plana o redonda sin que esto afecte su forma), mientras que los cambios en las reglas de un *código de un programa* implica un cambio inmediato en su comportamiento (como en el ejemplo del botón  donde el 'carácter prefijo' determina la longitud de la palabra).

Bajo la perspectiva de estas comparaciones, la enunciación de reglas en un *código de un programa* es similar a fabricar *leyes físicas* que no pueden ser cuestionadas y que están estrechamente relacionadas a la forma en que un programa va actuar.

De esta forma, los programas funcionan como *contextos* donde las reglas son sus *condiciones* y la vez las *causas* que determinan al programa (a diferencia de un *código penal*, por ejemplo, donde las reglas son *razones* de acción). Por ejemplo, podríamos decir que el código en *explorer.exe* plantea las *causas* por las cuales el botón  contiene ese ícono, está etiquetado con la palabra 'Inicio' y al ser oprimido se despliega un menú.

Estas *condiciones* son las que determinan los límites de lo *posible* en un programa. De forma similar a otros *contextos* como el mundo que habitamos<sup>6</sup>, dichos límites se aprenden por medio de la enseñanza, la relación establecida con otros contextos (movimiento, representación), o el ensayo y error. En la mayoría de los casos es posible que no se pueda predecir lo que va a suceder exactamente, sin embargo, al conocer los límites, podemos saber que NO va a suceder. Por ejemplo si jugamos ajedrez en un programa como Deep Fritz, lo más seguro es que no sepamos con exactitud cuál va a ser su próximo movimiento, pero, por lo menos, confiamos en que dicho movimiento no va a infringir las reglas de ajedrez que nosotros conocemos (no va a mover un peón como si fuera un alfil). En la medida en que se esté familiarizado con un programa, con mayor exactitud se sabrá que esperar o no de él.

---

<sup>6</sup> Se debe conservar cierta distancia al comprar un programa de computador con el mundo, ya que este último es mucho más complejo y compuesto por demasiadas variables y principios – hasta la mancha más pequeña puede afectar el trayecto de los eventos, mientras que en un programa de computador dicha mancha debe ser añadida.



Así que en el caso de los juegos para computador, las *reglas del juego* se convierten en *condiciones de contexto*. Por ejemplo, cuando dos personas juegan ajedrez en un tablero, aceptan que un peón sólo se puede mover hacia adelante y que un alfil sólo lo puede hacer en diagonal, a pesar de que no haya nada que físicamente les impida realizar otros movimientos o lanzar las piezas contra su adversario. Los jugadores acuerdan o auto-imponen las reglas – *suspenden* la legislación compleja y difusa del mundo ordinario y la remplazan por una con “reglas precisas, arbitrarias e irrecusables” (CAILLOIS).

Pero cuando la superficie en la que se juega es un computador, estas reglas no se aplican sólo porque existe un acuerdo entre sus jugadores (ni siquiera es necesario) sino por que están determinadas por las *condiciones* de este *contexto* (el programa). Si, por ejemplo, hay un jugador tramposo que siempre roba las piezas de su oponente, al jugar en el programa esto le será ‘físicamente’ imposible ya que, según el código del programa, esto no puede suceder (al igual que en nuestro mundo físico es imposible que un cuerpo material pueda atravesar una pared sin romperla).

En el caso del video juego de fútbol las leyes físicas que el programa emula, por ejemplo la gravedad que afecta al balón, son del mismo tipo que las reglas del juego (si el balón pasa la línea blanca de la portería es gol): ambas se comportan como ‘leyes naturales’. El jugador puede asumir o negar dichas leyes de la misma forma en que nosotros podemos asumir o negar que el fuego quema: si lo asumimos evitaremos tocarlo (el jugador jugará), si lo negamos lo más seguro es que nos quememos (cuando el balón entre a la portería inevitablemente será gol). Es decir, se anula el error de apreciación, como cuando un jugador le hace una falta a otro y el arbitro no la ve, o cuando recae en este la decisión de una circunstancia ambigua (entró o no el balón).

Ahora, no sólo las reglas del juego se transforman en ‘leyes naturales’ que rigen a estos *contextos* sino que los jugadores del programa (contra los que se juega cuando se dice “estoy jugando contra el computador”) también lo hacen, ya que su forma de jugar también está determinada por un código programático. Por lo tanto, este ‘jugador’ no es un cuerpo habitante de este *contexto*, sino otro elemento

pertenece a él. Por ejemplo, Deep Fritz no se entiende como jugador sino como un *componente del juego* encargado de analizar y realizar jugadas: es un efecto más de las causas determinadas por el código, lo que convierte a Deep Fritz en tanto el juego como el jugador - tablero y estrategia emergen como un único elemento.

Por lo tanto, cuando se dice que se juega contra un programa es similar a cuando se dice “compito contra la montaña” o “corro contra el reloj”. El contrincante al que se hace referencia no es un sujeto concreto que también esté jugando, sino un conjunto de condiciones del contexto que imponen cierta dificultad para alcanzar un objetivo. Más adelante hablaré de esto, por el momento debemos tener en cuenta que el contrincante, al ser parte del *contexto*, no puede ser *jugador*, y que por lo tanto un programa no juega – sólo sigue los parámetros que se le han establecido.

Por otro lado, siendo que el juego es una suspensión momentánea del mundo ordinario (en cuanto que los jugadores asumen por un tiempo y en un espacio limitado algunas reglas que no son determinadas por las leyes físicas o reglas sociales) sus reglas coercionan a la persona sólo cuando está jugando. Fuera del juego la persona tiene otra manera de actuar (no persiguen un balón todo el día) y seguramente dentro de otros ambientes haya otro tipo de reglas, implícitas o explícitas, que lo hagan comportar de cierta manera. Por el contrario, un programa de computador sólo está en capacidad de actuar dentro de los ambientes donde fue programado para hacerlo.

Esto no sólo se aplica a los juegos para computador, sino también para todas las aplicaciones y programas. La diferencia es que ya no los concebimos como jugadores sino como *agentes*, entendiendo esta palabra de forma similar a cuando decimos, por ejemplo, *agente de policía*: es una persona que lleva a cabo tareas específicas regidas por leyes y reglas que indican su intención y forma de actuar (hasta su forma de vestir). Además de estar hablando de un programa y no de una persona, la diferencia es que si el policía actuó fuera de estas normas ya no actuará como policía, mientras que un programa no tiene la capacidad de hacer cosas fuera de las reglas que lo determinan, es decir, no puede suspender *su* realidad.

Sin embargo no se puede pasar por alto que el diseño del programa (sus funcionalidades) ha sido establecido por la persona a cargo de su desarrollo y que, de una u otra forma, el programa las ha *heredado* sus intenciones y razones. En otras palabras: el programa esta *diseñado* para que actúe y cumpla con las *intenciones* de su programador, lo que hace de la *intención* algo implícito en la *acción* que lleva a cabo el programa, así este no sepa por qué hace lo que hace (porque al final de cuentas no sabe nada).

Otra característica del agente es que debe estar circunscrito dentro de unas coordenadas de relación con otros agentes, cosa que en los programas de computador no es extraño ya que éstos, por lo general, interactúan con otros programas y/o personas. Para que exista tal relación se requiere de ‘acoples’ que los unan a los diferentes agentes; algunos son acoples de entrada (input), otros son acoples de salida (output) y otros son bidireccionales.

En el caso de los acoples-*input* de un programa, los acoples programa→programa y humano→programa funcionan más o menos de la misma manera: aceptan *datos* que son expuestos a las *condiciones* de un programa que puede afectarlos y/o ser afectado por ellos (de forma similar cuando se expone el agua a las condiciones de un refrigerador y éste lo transforma en hielo).

Pero el caso de los acoples-*output* es diferente si se trata de un acople programa→programa o de un acople programa→humano: en el primer caso, el acople mantiene una continuidad mecánica (meramente funcional) cuya base es la relación *causa-efecto* (un modelo de engranajes). Este es el mismo tipo de relación que establece el humano con otro tipo de máquinas como la endoscopia, la respiración artificial o los medicamentos – todos se acoplan al cuerpo humano de manera mecánica y a un nivel biofísico. Pero cuando hablamos de relaciones programa→humano los acoples, por lo general, no son mecánicos sino representacionales (simbólicos) y el humano debe interpretar y construir *sentido* a partir de ellos.

Este tipo de relación es lo que permite mantener una retro-alimentación entre el usuario y el programa, ya que encuentra en el *output* del programa (el informe de

estado) una representación suya que responde a las decisiones que él toma, que al mismo tiempo le permite conocer los límites de lo posible en el programa.

# Eliza y el sentido

---

## **Eliza**

Eliza es un programa que fue desarrollado por Joseph Weizenbaum en el Massachusetts Institute of Technology (MIT) en 1966, con el objetivo de hacer “posible cierto tipo de conversación en lenguaje natural entre hombre y computador”<sup>7</sup>.

El programa consiste en que el usuario introduce un texto el cual Eliza inspecciona en búsqueda de *palabras-clave*. Si una de estas palabras es encontrada, el programa re-ensambla el texto según una regla que asociada con ella. Por ejemplo, si la frase *input* es “yo estoy trabajando” el programa encuentra las *palabras-claves* “yo estoy”, que se relacionan con una regla que indica que (1) se deben reemplazar las *palabras-claves* por “hace cuánto has estado x”, (2) se debe añadir la parte de la frase que no se identificó (en este caso “trabajando”), y (3) se debe añadir al final de la frase un signo de interrogación (uno al principio y otro al final en el caso del idioma español). Después de que se ha transformado la frase según estos parámetros, devuelve al usuario la frase resultante (*output*) “¿hace cuánto has estado trabajando?”.

Dichas reglas y su relación con las *palabras-claves* son extensibles y reemplazables. Cualquier persona que conozca el programa, tenga acceso al archivo que contiene dichas reglas (Weizenbaum lo llama *scripts*: archivos separados del programa encargado de la lógica) y cierta idea de cómo funcionan, puede modificarlas. Sin embargo, para fines prácticos, en este texto voy a referirme a las reglas que utiliza la versión del programa que aparece en <http://chayden.net/eliza/Eliza.shtml>, la cual mantiene los elementos básicos del programa original. También se debe tener en cuenta que estas reglas han sido escritas para el idioma inglés, cuya estructura gramatical y uso de palabras difiere del español: los textos que aparecen en este capítulo son adaptaciones (antes que traducciones) al español.

---

<sup>7</sup> Original en Inglés, traducción propia: “...which makes certain kinds of natural language conversation between man and computer possible.”

Men are all alike.

IN WHAT WAY?

They're always bugging us about something or other.

CAN YOU THINK OF A SPECIFIC EXAMPLE?

Well, my boyfriend made me come here.

YOUR BOYFRIEND MADE YOU COME HERE

He says I'm depressed much of the time.

I AM SORRY TO HEAR YOU ARE DEPRESSED

It's true. I am unhappy

DO YOU THINK COMING HERE WILL HELP YOU NOT TO BE UNHAPPY

I need some help, that much seems certain.

WHAT WOULD IT MEAN TO YOU IF YOU GOT SOME HELP

Perhaps I could learn to get along with my mother.

TELL ME MORE ABOUT YOUR FAMILY

My mother takes care of me.

WHO ELSE IN YOUR FAMILY TAKES CARE OF YOU

My father.

YOUR FATHER

You are like my father in some ways.

WHAT RESEMBLANCE DO YOU SEE

You are not very aggressive but I think you don't want me to notice that.

WHAT MAKES YOU THINK I AM NOT VERY AGGRESSIVE

You don't argue with me.

WHY DO YOU THINK I DON'T ARGUE WITH YOU

You are afraid of me.

DOES IT PLEASE YOU TO BELIEVE I AM AFRAID OF YOU

My father is afraid of everybody.

WHAT ELSE COMES TO YOUR MIND WHEN YOU THINK OF YOUR FATHER

Bullies.

DOES THAT HAVE ANYTHING TO DO WITH THE FACT THAT YOUR BOYFRIEND  
MADE YOU COME HERE

*Conversación utilizada por Weizenbaum para describir el funcionamiento de Eliza en el documento citado. Las líneas en mayúscula corresponden a Eliza. En <http://www.stanford.edu/group/SHR/4-2/text/dialogues.html> se encuentran varias conversaciones legendarias de Eliza y de otros programas diseñados para sostener conversaciones.*

Existen varias reglas, unas más complejas (como “tu me x a mí” debe ser remplazado por “¿por qué yo te x a tí?”) y otras más sencillas (como “mi x” debe ser remplazado por “tu x”), y otras que no especifican una transformación de la frase, sino una respuesta estática a ciertas palabras. Por ejemplo, cuando el usuario utiliza la palabra ‘parecido’ (sin importar la manera en la que sea usada) Eliza responde con un “¿y dónde ves tú la semejanza?”.

Cuando no encuentra ninguna palabra clave responde con frases comodín que muestran cierto interés en la conversación, como por ejemplo: “Por favor, continúa.”, “Podrías elaborar un poco más” o “¿Eso qué significa?”.

El “¿cómo hacer posible una conversación?” es en realidad una pregunta por cómo poder manipular textos de tal forma que se pueda mantener coherencia sin saber de que se está hablando. En Eliza esto se intenta, en algunos de los casos, transformando la frase, que el usuario escribe, en una pregunta. En todos los casos lo que se evita es dar *razón* de algo. Por ejemplo, hay una regla que establece que cuando una frase termina en signo de interrogación, por ejemplo: “¿qué día es hoy?”, Eliza contesta con “¿eso es una pregunta o una afirmación?”, “¿te interesa saberlo?” o “¿en realidad crees que puedo responderte qué día es hoy, cierto?” - las repuestas siempre *evitan* que Eliza deba saber algo sobre lo que se le pregunta. Otra estrategia de evitación es responder a la pregunta “¿tu x?” (la cual podría ser referente a alguna información de Eliza como por ejemplo “¿tu tienes alma?”) con “Estamos hablando acerca de tí, no de mí”.

Como se notará, la temática de las conversaciones de Eliza es el usuario - no se habla ni de Eliza, ni del clima, ni de política, ni de fútbol. Aunque el usuario quisiera mencionar estas temáticas, Eliza respondería con una pregunta referente al usuario o con una muletilla para inducirlo a que continúe hablando hasta que pueda encontrar el quiebre y volver a centrar la conversación en él. Un ejemplo simple: El usuario dice “Está lloviendo”, Eliza responde “Sígueme contando”, el usuario contesta “No me gusta cuando llueve” a lo cual Eliza responde con “¿Por qué no te gusta cuando llueve?”.

Wiezenbaum nos indica que el comportamiento de Eliza es similar a de algunos psicoterapeutas<sup>8</sup>. Él sustenta que este tipo de conversación es de los únicos donde:

[...] uno de los dos participantes es libre de asumir la postura de no conocer casi nada sobre el mundo real. Si, por ejemplo, uno le dijera a un siquiatra, “Fui a un viaje largo en bote” y él responde “Hábleme sobre los botes”, uno no asumiría que no sabe nada acerca de botes, sino que tiene algún propósito en dirigir la conversación subsecuente. Es importante tener en cuenta que esta suposición es hecha por el *hablante*<sup>9</sup>. [...] En cualquier caso, esto tiene una utilidad psicológica crucial en cuanto que sirve al *hablante* a mantener su sensación de ser oído y entendido. El *hablante* defiende su impresión (que aún en la vida real puede ser ilusoria) atribuyendo a su pareja de conversación todo tipo de conocimiento sobre las generalidades, ideas y habilidad de razonamiento. Pero de nuevo, estas son las contribuciones que hace el *hablante* a la conversación. (WEIZENBAUM)<sup>10</sup>

Es de resaltar que Weizenbaum llame ‘hablante’ sólo al paciente y no al terapeuta. No podría dar la razón exacta de por que él escogió esta palabra (la cual además es protagonista en esta cita) sin embargo podríamos sacar algunas conclusiones si tenemos en cuenta que el terapeuta (en la forma como lo describe Weizenbaum) es pasivo en la conversación: la intención (el sentido que se hace) de la pregunta es asumida por el ‘hablante’, y este es el único que expresa algo con sentido en esta conversación. Sin embargo en este escrito he decidido cambiar la palabra ‘hablante’ por ‘usuario’ ya que esta última especifica a una persona que se enfrenta con un sistema digital y mantiene el sentido de dicha relación.

Por ejemplo el programa Eliza no hace *sentido* de lo que se le está diciendo ni de lo que dice. Este recae totalmente en lo que el usuario asume (ya sea como ilusión o como realidad) de la conversación y del programa. Siendo así, Eliza no

8 Weizenbaum establece que Eliza fue un programa inspirado en los psicoterapeutas que pertenecían a la escuela ‘rogeriana’. En este texto no voy a entrar en detalles de dicha escuela, sin embargo cabe resaltar que uno de los ejes de este método de terapia era simplemente guiar al usuario a través de una autocomprensión del problema, es decir el terapeuta no le daba argumentos ni explicaciones sino que sólo motivaba al cliente a seguir hablando.

9 La palabra utilizada por Weizenbaum es *speaker* la cual puede ser traducida como orador, interlocutor o simplemente hablante. En todos los casos es la persona que expresa algo.

10 Original en Ingles, traducción propia: “[...] one of the participating pair is free to assume the pose of knowing almost nothing of the real world. If, for example, one were to tell a psychiatrist "I went for a long boat ride" and he responded "Tell me about boats", one would not assume that he knew nothing about boats, but that he had some purpose in so directing the subsequent conversation. It is important to note that this assumption is one made by the speaker. [...] In any case, it has a crucial psychological utility in that it serves the speaker to maintain his sense of being heard and understood. The speaker further defends his impression (which even in real life may be illusory) by attributing to his conversational partner all sorts of background knowledge, insights and reasoning ability. But again, these are the speaker's contribution to the conversation.”



necesita tener ningún tipo de *conocimiento*, no necesita *saber que* (saber acerca de botes) sino *saber como* (cómo responder a una frase que habla de botes) – los programas, los computadores y las máquinas en general son sobre todo entes de acción - que *saben cómo hacer algo*. El *conocimiento* en ellos se traduce a datos (que a su vez para nosotros se transforman en información) que no necesitan comprender, sólo guardar, jalar, empujar, manipular, comparar, eliminar y hasta fabricar.

### ***No soy yo, eres tú***

En uno de mis primeros experimentos, *No soy yo, eres tú*, decidí hacer uso de Eliza. La idea de este ejercicio era simple: poner a Eliza en mi lugar en un programa de mensajería instantánea (MI) llamado Microsoft Messenger que permite conversar con personas a través de Internet. En otras palabras, puse a Eliza a hablar por mí.

El punto de partida de este trabajo (junto a otros ejercicios que hice) era interrumpir la forma en que se presenta y conforma la *identidad* dentro de los sistemas de auto-identificación que abundan en Internet. Acá el término 'identidad' no hace referencia a conceptos metafísicos ni culturales, sino meramente cognoscitivos (Juan está sentado en la silla, pero tanto él como la silla tienen una identidad propia e independiente). La intuición que sirvió como punto de partida del ejercicio (no podría decir que es una teoría ni una hipótesis) fue encontrar que hay medios de comunicación donde la expresión (lo que se comunica) no está separada de la identidad, y que hay otros donde la expresión se aísla de esta identidad conformándose como objeto independiente. Por ejemplo, al hablar con otra persona por teléfono se trasmite la voz de las personas en directo, lo cual mantiene una relación de identidad-expresión estrecha, ya que esta expresión más que ser un producto es una acción (como respirar por ejemplo). Por el contrario, una carta es una expresión que se separa del sujeto que la escribe y se convierte en un producto suyo - un objeto aparte - (la mayoría de las expresiones escritas mantienen esta relación con el sujeto: reportajes, novelas, ensayos, etc.; son productos que se separan del sujeto – identidad - que los escribió). Pero cuando hablamos de MI (sólo por nombrar un caso), a pesar de que la expresión escrita no tiene esa relación



*No soy yo, eres tú. Ventana de conversación en Microsoft Messenger.*

estrecha con el sujeto (no hay voz, no hay cuerpo, la expresión no es un efecto), lo que se escribe no se considera un producto intermediario (objeto aparte).

Por otro lado, también se basaba en la premisa de que los medios de comunicación son sistemas que mantienen cierto nivel de transparencia en lo que transmiten. Es decir, el *input* que se les introduce es enviado al receptor con muy poca variación. Ese nivel de opacidad (el hecho de que haya niveles de 'pureza' en el mensaje) es importante: en medios como la radio, el teléfono o la televisión hay ruido que opaca (en un bajo grado por lo general) el mensaje. Por el contrario la carta y los mensajes por Internet mantienen un nivel de transparencia superior: la carta enviada a través del correo postal llega intacta (de lo contrario sería un pésimo servicio), y el mensaje enviado por MI llega sin alteración alguna (como en muchos sistemas digitales).

Eliza es el contra-ejemplo de estos sistemas: el *input* que se le introduce es alterado, transformado y devuelto de una forma totalmente diferente a como se introdujo. El papel de Eliza dentro de *No soy yo, eres tú* entonces era volver opaca una conversación que se pretende transparente, como también (ya lo mencioné) en establecer la dicotomía ente-producción dentro de este medio de comunicación (MI).

Esto implica que las personas con las que entablaba conversación no sabían que se trataba de un sistema opaco que funcionaba como un espejo de feria que distorsiona todo lo que se le dice - el sentido de la conversación recaía en el usuario engañado. Por esta razón decidí nombrar al ejercicio *No soy yo, eres tú* que, además de ser el título del trabajo, también era el nombre de pantalla (apodo, nickname), dándole una pista a las víctimas de que no estaban conversando conmigo, sino que (de una forma indirecta) con ellos mismos.

Lo que acabo de describir someramente fue el punto de partida de este ejercicio, sin embargo lo que influyó en el curso de mi investigación no fue esta base sino el resultado obtenido de el ejercicio. Es decir, influyó más como experimento que como intervención: la naturaleza abierta del ejercicio (la dependencia de una participación no voluntaria pero también no predecible de terceros) dejaba muchas brechas que se traducían en un 'a ver que pasa' - en un experimento.

En las siguientes páginas se encuentran algunas conversaciones que he escogido y que me llevaron a sacar ciertas conclusiones. Las he organizado por orden de ilusión y coherencia en la conversación. (Advertencia: las conversaciones se presentan en este documento tal cual sucedieron, es decir: con errores ortográficos y gramaticales como también con convenciones propias de una escritura informal):

## 1. sergio

No soy yo, eres tú: y eso que significa?

sergio: que??????

No soy yo, eres tú: te interesa?

sergio: mmm??????

No soy yo, eres tú: Sientes la necesidad de discutir eso?

sergio: discutir que????????

No soy yo, eres tú: es una pregunta o una afirmacion?

sergio: es una afirmacion

No soy yo, eres tú: sigueme contando

sergio: 1, 2, 3, 4, 5, 6, 7, 8, .....

No soy yo, eres tú: y eso que significa?

sergio: te estoy contando

No soy yo, eres tú: es por que tu estas contando que viniste a mi?

sergio: jamaz he ido a ti

No soy yo, eres tú: Sientes la necesidad de discutir eso?

sergio: no

No soy yo, eres tú: estas diciendo no por que si?

sergio: si

No soy yo, eres tú: crees que es probable que ?

sergio: no

No soy yo, eres tú: eso es negativo

sergio: asoweiiiiiiiiiiiiii

No soy yo, eres tú: sigueme contando

sergio: que

No soy yo, eres tú: nose

sergio: bueno me parece que estas loco

No soy yo, eres tú: en serio te parece que estabas loco ?

sergio: usted

No soy yo, eres tú: estabamos discutiendote a ti -- no a mi.

sergio: no estas equivocado

No soy yo, eres tú: quisieras creer que yo estava equivocado ?

No soy yo, eres tú: Camilo te manda saludos. Hablas con Eliza.

sergio: crees que vale la pena discutirlo

No soy yo, eres tú: te interesa?

sergio: ?

No soy yo, eres tú: y eso que significa?  
 sergio: es una pregunta o una afirmacion  
 No soy yo, eres tú: Sientes la necesidad de discutir eso?  
 sergio: dimelo tu  
 No soy yo, eres tú: oh, yo ?  
 sergio: si  
 No soy yo, eres tú: deseas que ?  
 sergio: eliza la que me lo.....  
 No soy yo, eres tú: de que forma te lo ?  
 sergio: termina la frase  
 No soy yo, eres tú: sigueme contando  
 sergio: camilo quien es eliza  
 No soy yo, eres tú: y eso que significa?  
 sergio: camilo quien es eliza?  
 No soy yo, eres tú: Sientes la necesidad de discutir eso?  
 sergio: camilo quien es eliza?  
*(sergio pregunta una y otra vez "camilo quien es eliza?" hasta que desemboca en:)*  
 sergio: camilo quien es elizacamilo quien es elizacamilo quien es elizacamilo  
 quien es elizacamilo quien es elizacamilo quien es elizacamilo quien es  
 elizacamilo quien es elizacamilo quien es elizacamilo quien es elizacamilo quien  
 es elizacamilo quien es elizacamilo quien es elizacamilo quien es elizacamilo  
 quien es elizacamilo quien es elizacamilo quien es elizacamilo quien es  
 elizacamilo quien es elizac  
 No soy yo, eres tú: y eso que significa?  
*(sergio continua con la misma pregunta y la conversaci3n termina en:)*  
 sergio: fuck you

## 2. Mateo ®

No soy yo, eres tú: y eso que significa?  
 Mateo ®: eso?  
 No soy yo, eres tú: Sientes la necesidad de discutir eso?  
 Mateo ®: que es eso?  
 No soy yo, eres tú: te interesa?  
 Mateo ®: uh?  
 No soy yo, eres tú: sigueme contando  
 Mateo ®: hehe virus  
 No soy yo, eres tú: y eso que significa?  
 Mateo ®: wa  
 Mateo ®: wa  
 Mateo ®: wa  
 Mateo ®: wa  
 Mateo ®: wa  
 No soy yo, eres tú: Sientes la necesidad de discutir eso?  
 No soy yo, eres tú: y eso que significa?

No soy yo, eres tú: sigueme contando  
 Mateo ®: haha  
 No soy yo, eres tú: sigueme contando  
 No soy yo, eres tú: y eso que significa?  
 Mateo ®: spam  
 No soy yo, eres tú: Sientes la necesidad de discutir eso?  
 No soy yo, eres tú: Camilo te manda saludos. Hablas con Eliza.  
 Mateo ®: hahaha lo mismo =)

### 3. [b]noSound[/b]

No soy yo, eres tú: y eso que significa?  
 [b]noSound[/b]: que?  
 No soy yo, eres tú: te interesa?  
 [b]noSound[/b]: jajaja no entiendo jajajaja  
 No soy yo, eres tú: estas diciendo no por que si?  
 [b]noSound[/b]: de nuevo que significa que? Je  
 No soy yo, eres tú: es una pregunta o una afirmacion?  
 [b]noSound[/b]: aish! me perdi de nuevo jajaja  
 No soy yo, eres tú: en serio te perdi de nuevo jajaja ?  
 No soy yo, eres tú: Camilo te manda saludos. Hablas con Eliza.  
 [b]noSound[/b]: je pues no es comun que se inicien las conversaciones asi je, hola eliza quien eres? jajaja dile a cami que lo mismisimo  
 No soy yo, eres tú: eso es negativo  
 [b]noSound[/b]: :hum neutral  
 No soy yo, eres tú: Sientes la necesidad de discutir eso?  
 [b]noSound[/b]: no eres la que preguntas  
 No soy yo, eres tú: por que no?  
 [b]noSound[/b]: porque es falso  
 No soy yo, eres tú: es esa la razon real?  
 No soy yo, eres tú: Camilo te manda saludos. Hablas con Eliza.  
 [b]noSound[/b]: es un imaginario!!!  
 No soy yo, eres tú: sigueme contando  
 [b]noSound[/b]: llego mi pizza!!  
 No soy yo, eres tú: tu pizza ?  
 [b]noSound[/b]: afirmas?  
 No soy yo, eres tú: pero tu pizza .  
 [b]noSound[/b]: es femenino o masculino?  
 No soy yo, eres tú: y eso que significa?  
 [b]noSound[/b]: lo que tu preguntes  
 No soy yo, eres tú: nose  
 [b]noSound[/b]: listo para mañana?  
 No soy yo, eres tú: Sientes la necesidad de discutir eso?  
 No soy yo, eres tú: Camilo te manda saludos. Hablas con Eliza.

#### 4. Lorena al extremo

No soy yo, eres tú: y eso que significa?

Lorena al extremo: que cosa?

No soy yo, eres tú: te interesa?

Lorena al extremo: de que hablas?

No soy yo, eres tú: es una pregunta o una afirmacion?

No soy yo, eres tú: Camilo te manda saludos. Hablas con Eliza.

Lorena al extremo: cami tu como que te fumaste algo

No soy yo, eres tú: estabamos discutiendote a ti -- no a mi.

Lorena al extremo: ujsajuas ok

No soy yo, eres tú: Sientes la necesidad de discutir eso?

Lorena al extremo: quien es eliza, yo no me acuerdo

No soy yo, eres tú: piensas seguido en ?

Lorena al extremo: porque me estas diciendo eso?

No soy yo, eres tú: en serio te estabas diciendo eso ?

Lorena al extremo: //que?

No soy yo, eres tú: sigueme contando

Lorena al extremo: mejor me voy

No soy yo, eres tú: de que forma te voy ?

No soy yo, eres tú: Camilo te manda saludos. Hablas con Eliza.

Lorena al extremo: camilo ponte serio

No soy yo, eres tú: y eso que significa?

Lorena al extremo: quien es eliza?

No soy yo, eres tú: Sientes la necesidad de discutir eso?

Lorena al extremo: pues al citar un nombre propio si.

No soy yo, eres tú: no estoy interesado en nombres

Lorena al extremo: esto tiene el tono de una conversacion para pensar

No soy yo, eres tú: sigueme contando

Lorena al extremo: pero ahora no puedo porque estoy haciendo un ensayo sobre la caida y la sexualidad

No soy yo, eres tú: como asi que no puedes porque estoy haciendo un ensayo sobre la caida y la sexualidad ?

Lorena al extremo: si. no puedo pensar mas alla de eso

No soy yo, eres tú: crees que es probable que ?

No soy yo, eres tú: Camilo te manda saludos. Hablas con Eliza.

Abran notado la frase “Camilo te manda saludos. Hablas con Eliza”. Este mensaje lo programé para que apareciera después de unos minutos de que la otra persona no respondiera, con el objetivo de reactivar la conversación (resultó siendo bastante efectivo).

Como se aprecia en todas las conversaciones hay una gran confusión por parte de los usuarios (llamaré así a las personas con las que conversó Eliza) y cada uno tuvo un acercamiento diferente a la hora de enfrentarla o, para decirlo

reinterprentando la aserción de Weizenbaum: cada usuario intentó encontrar el sentido de lo que sucedía de diferentes formas.

En las dos conversaciones que prosperaron (la 3 y la 4) los usuarios intentaban hacer sentido de lo que se decía sin desfallecer en pensar que en realidad se trataba de mí. 'Lorena al extremo' se preguntaba si yo "me había fumado" y se angustiaba por la conversación. La ilusión en el caso 3 no duró tanto: se levantó una sospecha que desembocó en un "es un imaginario!!!" (nunca supe a qué se refería). Sin embargo al principio intentó encontrar el sentido de la forma más normal: preguntando y manifestando su confusión.

En el caso 2 el encontrarle sentido a lo que sucedía no tuvo tantos rodeos: se atribuyó este extraño comportamiento a un virus (informático). Y en el caso 1 el usuario se mantuvo preguntándome quién era Eliza tanto a forma de test (forma de probar que sucede) como de exigencia para que le explicara qué sucedía.

Digamos que de una u otra forma el ejercicio sí logró imponerse como opacidad en un medio que se pretende transparente y desligar a la persona (yo en este caso) de la identidad dentro del medio. Esto era lo que se esperaba que sucediera y no era muy difícil de predecir. Sin embargo lo que emergió fueron situaciones de mal entendidos, confusión y frustración (elementos propios de la insistencia de Eliza de evitar hablar de otra cosa que no sea del usuario), donde imperaba el intento de encontrar una razón (o por lo menos una causa) de lo sucedía.

El hecho de ver que Eliza funciona no como un programa que permite la conversación en lenguaje natural entre un computador y una persona, sino más bien más como un programa que niega al usuario y sabotea su confianza en la predictibilidad de la retro-alimentación, lanzó a mi investigación a fijarme en aquellas situaciones donde el usuario es negado por el computador, donde se boicotea el acople programa→humano correspondiente a lo representacional y lo comprensible por una persona. Pero de esto hablaré en el último capítulo de este texto, antes debo introducir otros ingredientes también influyentes en mi trabajo.



# Parásitos

---

Encontramos una resonancia en la filosofía de Félix Guattari al establecer que las máquinas no saben qué sino que saben cómo:

Aquí se impone una distinción entre las semiologías productoras de significantes moneda común de los grupos sociales-, como la enunciación “humana” de las personas que trabajan alrededor de la máquina y, por otra parte, semióticas asignificantes que, más allá de la cantidad de significados que vehiculicen, manejan figuras de expresión que podríamos calificar de “no humanas”; ecuaciones, planos que enuncian a la máquina y la hacen actuar a título diagramático sobre los dispositivos técnicos y experimentales. Las semiologías de la significación juegan sobre tableros de oposiciones distintivas de orden fonemático o escritural que transcriben los enunciados en materias de expresión signifiante. (GUATTARI)

Por un lado, esta división entre las semiologías productoras de significantes y las semiótica a-significantes divide el decir del hacer, en otras palabras: la ‘enunciación humana’ está fuertemente arraigada a la representación, a la mimesis, y, en muy pocas circunstancias<sup>12</sup> afecta directamente a lo que hace referencia ya que el lenguaje es simbólico, hace referencia a objetos y actos fuera de sí mismo. Las semióticas a-significantes no funcionan a través de las referencias, de algo externo a ellas, sino que son los mismos ‘signos’ los que se ven afectados por su manera de ‘enunciar’.

Lo que encontramos entonces son relaciones de afectación, no sólo en los signos, sean estos o no significantes, sino también entre máquina y máquina. Sin embargo para que haya tal afectación son necesarios los ‘acoples’ pertinentes para ellas. A continuación expongo tres casos de lo que yo consideraría parásitos, donde su forma de persistir es a través de las otras ‘máquinas’ que afecta:

En los pisos arenosos de ciertos lagos británicos, los camarones *gammarus insensibilis* se esconden de los pájaros que hacen de ellos un festín. Afortunadamente, para los predadores, algunos de estos camarones son más fáciles de cazar, ya que salen de su escondite en búsqueda de luz. Este tipo de comportamiento es contradictorio y va en contra de su supervivencia (están entrando directamente a la boca del lobo – o más bien del pájaro). Lo que lo provoca es una

‘enfermedad’: los camarones son ‘poseídos’ por un endoparásito (se *hospeda* en el interior del cuerpo) cuyo nombre científico es *microphallus papillorobustus*. Este es un gusano que ‘controla’ a su anfitrión (el camarón) y altera su comportamiento habitual.

La ‘razón’ por la cual el parásito incentiva al camarón a ser comido es por que de ello depende su supervivencia: el pájaro es su *anfitrión definitivo* (su lecho de muerte) donde madura y se reproduce. Dentro del crustáceo este no es más que una larva, que al pasar al tracto digestivo del pájaro se reproduce depositando sus huevos en el agua a través del excremento de su nuevo anfitrión. Estos huevos son comidos por los crustáceos, cerrando el complejo ciclo de vida de este parásito.

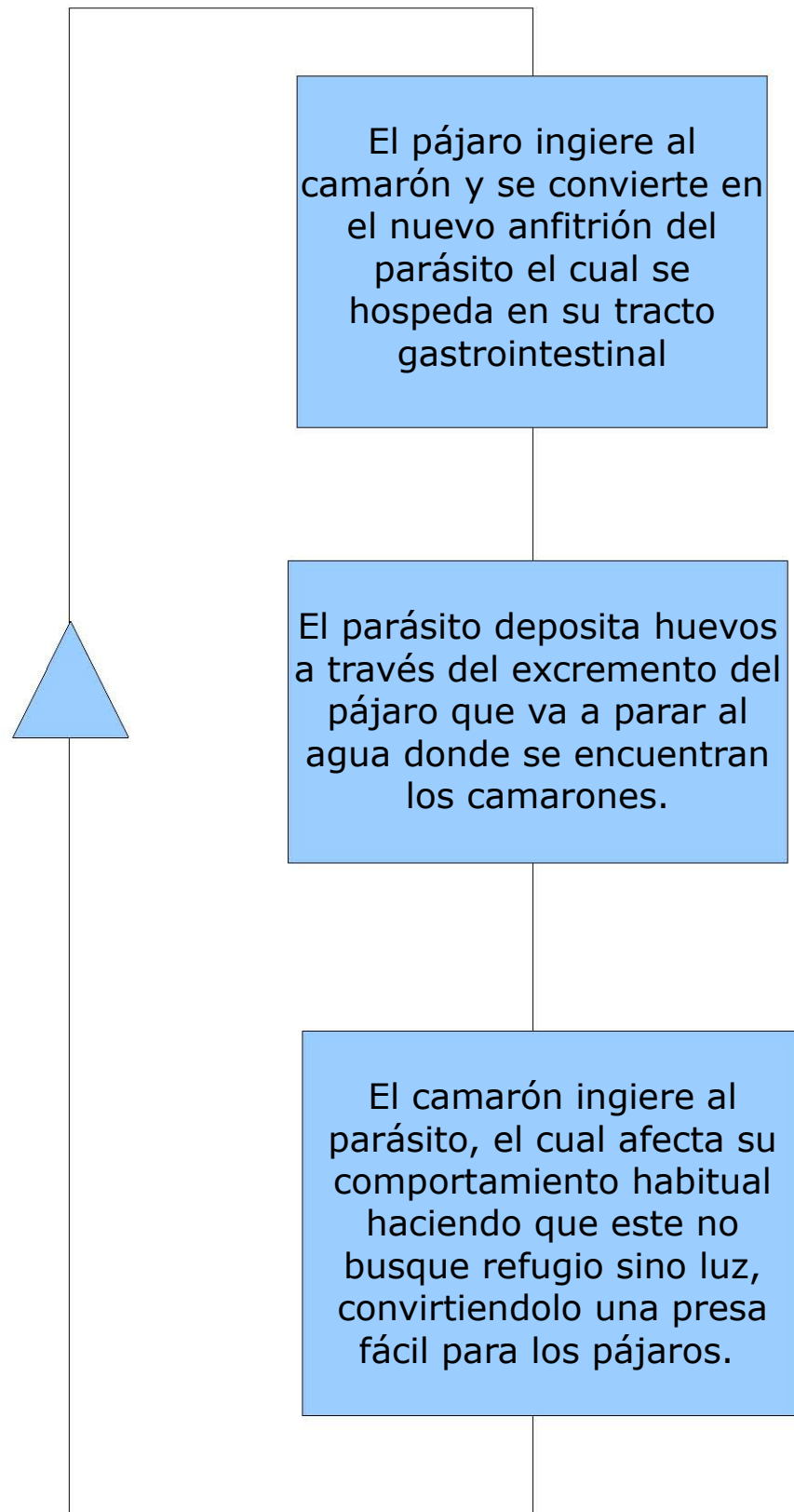
Resalta en esta descripción lo estratégico que es el gusano: afecta el comportamiento del crustáceo (hace que busque la luz y no el refugio) para que este sea una presa fácil, aumentando así las posibilidades de encontrar un anfitrión definitivo donde puede ‘vivir su etapa de pensionado’ y ‘hacer una familia’.

Los cuerpos del pájaro y el crustáceo son el *medio ambiente* del gusano, no sólo por que son las condiciones ambientales a las que el parásito está expuesto, sino por que también (de una forma más literal) son el *medio* por el cual él se *transmite*. Desde este punto de vista, el parásito es el *contenido* de este *medio* – del pájaro y del crustáceo.

Así tanto el pájaro como el camarón no saben que *transmiten* o *contienen* algo, simplemente se están alimentando. Necesitan ser ‘engañados’ y actuar como siempre lo hacen para que sean parte del proceso del parásito.

Podríamos decir que existe un comportamiento troyano: el ejercito se introduce en una fachada (el parásito habita al camarón) para llegar a un destino final (el pájaro) de forma desapercibida. También podríamos analizar un poco más a fondo esta analogía: el Caballo de Trola, a pesar de ser una estructura inerte, era una máquina con una intención: contenía un deseo heredado del ejercito constructor cómo también se engranaba con los deseos de la fortaleza enemiga que veía en él un regalo – este dispositivo contenía una *meta-subjetividad* conformada por ambos bandos (de no ser así no hubiera sido un artefacto efectivo). De la misma forma los

## Ciclo de vida del parásito m. papillorobustus



anfitriones del parásito son animados por una *meta-subjetividad* independiente de cualquier tipo de subjetividad que puedan tener. Lo vemos de una forma más radical en el caso del camarón cuyo comportamiento es alterado por el gusano. Pero esto no significa que los animales no infectados sean eximidos de esta maquinaria, sino que por el contrario son posibles anfitriones y, por lo tanto, entran en forma de negativo al ciclo de vida del parásito.

Así que desde el momento en que los (posibles) anfitriones son “engañados” (y desde el punto de vista del parásito) estos ya no son vistos como elementos primarios, sino como cuerpos necesarios para transmitir un *contenido* – como elementos vacíos ideales para mantener una información circulando (un *medio*).

Podemos asociar este caso con un ejemplo del arte: *Inserciones en circuito ideológico* de 1970. Este es un trabajo del artista brasilero Cildo Meireles quien pegaba el siguiente escrito en las las botellas de Coca-Cola:

*(Inglés original)*

Yankees go home!

INSERTION INTO

IDEOLOGICAL CIRCUITS

1 – Coca-Cola project

To register informations

and critical opinions on

bottles and return them

to circulation.

C. M.

*(Traducción al español)*

¡Yankees váyanse a casa!

INSERCIÓN EN

CIRCUITOS IDEOLOGICOS

1 – Proyecto Coca-Cola

Para registrar información

y opiniones criticas sobre

botellas y devolverlas

a circulación.

C. M.

Cuando estas estaban vacías, el escrito (de color blanco y letra chica) pasaba desapercibido en la fabrica donde re-embotellaban la gaseosa. Una vez estaban llenas el escrito resaltaba sobre el fondo negro del liquido y era distribuido junto con el producto (como los huevos del gusano se depositan en el agua a través del



*Inserción en circuitos ideológicos. Cildo Meireles. 1970.*

excremento del pájaro). En este caso, un ciclo industrial propio de la botella retornable se convierte en el *medio* del escrito de Meireles.

La frase “Yankees go home!” que hace parte del escrito es de uso común dentro de los círculos ‘anti-imperialistas’ que en los 70s tenían una gran acogida. Se podría decir que era (y es) un lugar común, llegaba a ser hasta un cliché. Pero la relación entre esta frase, la botella y la marca Coca-Cola, genera un significado en negativo (Yankees go home! vs ¡Que vivan los Estados Unidos!) que evidencia un medio y un mensaje dentro de maquinaria de circulación y consumo propia del capitalismo. Esto supone que la botella de Coca-Cola siempre fue un medio con un mensaje concreto, pero es a través de la intervención que hace Meireles que estos se hacen evidentes, lo que conlleva a que todas las botellas, a pesar de no tener el mensaje escrito, empiecen a leerse como un medio.

Vemos acá que, al igual que en el caso del parásito, también existe un *medio ambiente*: la acción de Meireles se establece dentro de un contexto puntual *mediático*, compuesto de divergencias e imaginarios. La frase “Yankees go home!” hace parte de estos imaginarios y la forma como circula es a través de un medio preexistente. Al ser la frase un imaginario, está se establece como estrategia de difusión (es el gancho), lo que en un principio crea una contradicción: lo que se comunica no puede ser la misma estrategia de comunicación. Pero es la acción (el intervenir un ‘circuito ideológico’) la que se establece como mensaje - lo que se comunica a través de la botella.

Esto se resalta en la otra parte del escrito que funciona como ‘partitura’ y validador de la inserción: es un auto-referente puesto en circulación. En este caso (y bajo un juego propio del arte conceptual) el mensaje sí es el mismo contenido. Pero además de esto, al ser ‘partitura’, hay una intención de que la acción sea ejecutada por otra persona. Esto se hace más fuerte si se tiene en cuenta que existen unas imágenes que apoyan esta intervención mostrando el ‘paso-a-paso’ para realizarla.

Así que el estado ideal sería que esta acción tuviera eco en otras personas y que ellas repitieran el procedimiento. De esta forma si las botellas con el escrito salen de circulación llegan otras a tomar su lugar.

Podemos hacer un paralelo entre esta intervención y el ciclo del gusano: hay un anfitrión intermediario que es la botella (el camarón en el caso del gusano) . El escrito cambia su 'comportamiento' (resalta su mensaje) lo cual se vuelve *atractivo* para un individuo que, en este momento, se convierte en el anfitrión definitivo (el pájaro en el caso del gusano), y donde el 'parásito-idea' llega a su 'madurez'. En este ambiente (dentro de los pensamientos del individuo) existe la posibilidad de que se reproduzca la acción, 'infectando' a otras botellas.

La diferencia con el parásito biológico es que este siempre es el mismo y nunca cambia su naturaleza a pesar de que habite varios cuerpos. En el caso del mensaje de Meireles este debe hacer una especie de *miosis metamórfica* y permanecer en ambos medios de diferente forma (y hasta trascender a otros): escrito en botella, idea en cabeza.

Si lo pensamos así, el ente que necesita sobrevivir es la acción originada por Meireles, donde los otros dos cuerpos (la botella y el individuo) sólo se comportan de tal forma que hagan posible su existencia: la botella infectada: al ser re-embotellada y comercializada; y el individuo al ver en esta acción algo que merece replicarse. Estos serían las dos condiciones principales para que se mantenga una reproducción constante.

Sin embargo debemos aceptar que esta no fue ni es la condición de este trabajo. Es muy posible que esta acción no haya sido replicada por nadie, o que simplemente no lo hizo con suficiente fuerza para 'sobrevivir'. El trabajo como un parásito-idea no sucedió, o desfalleció rápidamente. Varias condiciones (condiciones ambientales) pudieron intervenir en este proceso (los cuales no voy a discutir acá), y sin embargo es uno de los trabajos más influyentes de la segunda mitad del siglo XX. Esto quiere decir que, como gran parte de las obras de arte (sin importar su naturaleza), trascendió cómo *idea*, lo cual tiene una ventaja: se adapta, muta e interrelaciona con y a otras ideas generando nuevas producciones (algunas parasitarias, otras no).

Ahora hablemos de un caso de un programa digital: otro gusano pero esta vez no diseñado por la selección natural sino por alguien en Filipinas donde las leyes de

crímenes informáticos no existían en ese momento. Hablo del parásito digital *ILOVEYOU*, que fue una pesadilla en los noventas y trajo consigo altas pérdidas de información y dinero.

Este gusano consistía en un programa que, para reproducirse, tomaba la lista de direcciones de correo electrónico del computador infectado y enviaba un correo a cada una diciendo “por favor lee mi carta de amor”. Adjunto contenía un archivo llamado *carta-de-amor-para-ti.txt.vbs*<sup>11</sup>. El uso de una doble extensión (vbs y txt) era una estrategia para hacer pensar que se trataba de un archivo de texto cuando en realidad era un ejecutable. Una vez la gente lo abría el ciclo se repetía: se tomaban las direcciones de correos y se enviaba este mensaje. Una vez completaba esta tarea realizaba daños serios en los archivos de sistema del computador.

Encontramos entonces una situación parecida a la de las botellas de Coca-Cola de Cildo Meireles y a la del gusano *microphallus papillorobustus* (nuestro gusano biológico): hay un ente que, para poder subsistir, debe ‘habitar’ en el interior de dos cuerpos diferentes. En este caso es el computador el anfitrión destino (donde madura y se reproduce), y el humano que es el anfitrión intermediario.

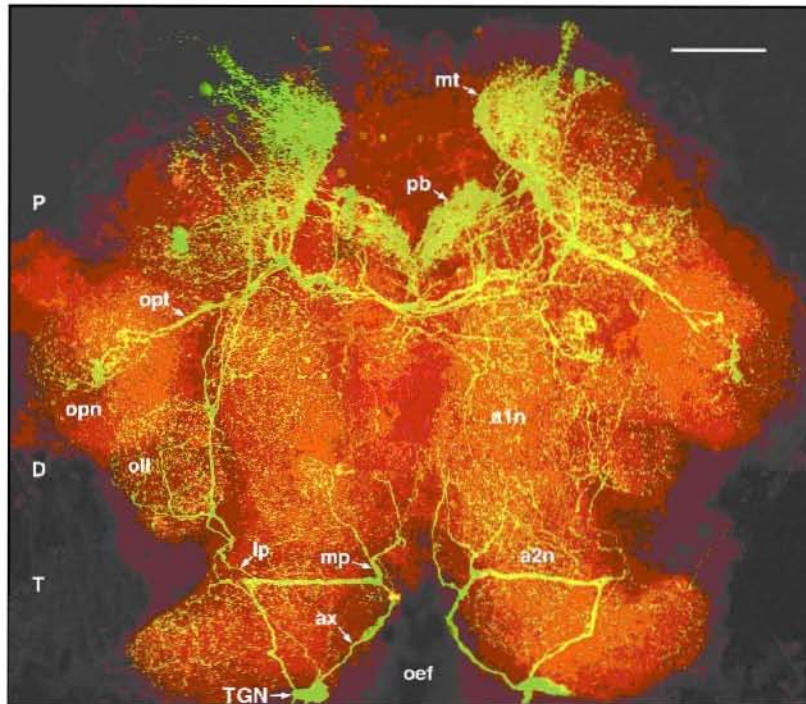
*ILOVEYOU* invade el cuerpo humano en forma de idea-gancho (al igual que el escrito de Meireles) y sólo es por medio de este proceso que nuestro gusano *ILOVEYOU* se puede reproducir. Bajo la forma de idea, el gusano seduce al individuo indicándole que hay un mensaje que él debe leer y que, en términos prácticos, hay un botón al que le debe hacer click. Al igual que el gusano biológico, *ILOVEYOU* controla a su anfitrión intermediario activando un comportamiento que es casi que una reacción automática. A pesar de que no fui víctima de este gusano me puedo imaginar las ansias con que la gente abría este correo sin importar quien lo mandara: su misma pareja, un familiar, un colega o hasta una persona desconocida.

El siguiente paso de este gusano era infectar el computador en el cual el usuario presiona el botón. Toma las direcciones, envía los correos, daña el computador. Aun siendo un procedimiento de detalles oscuros para el profano, esta

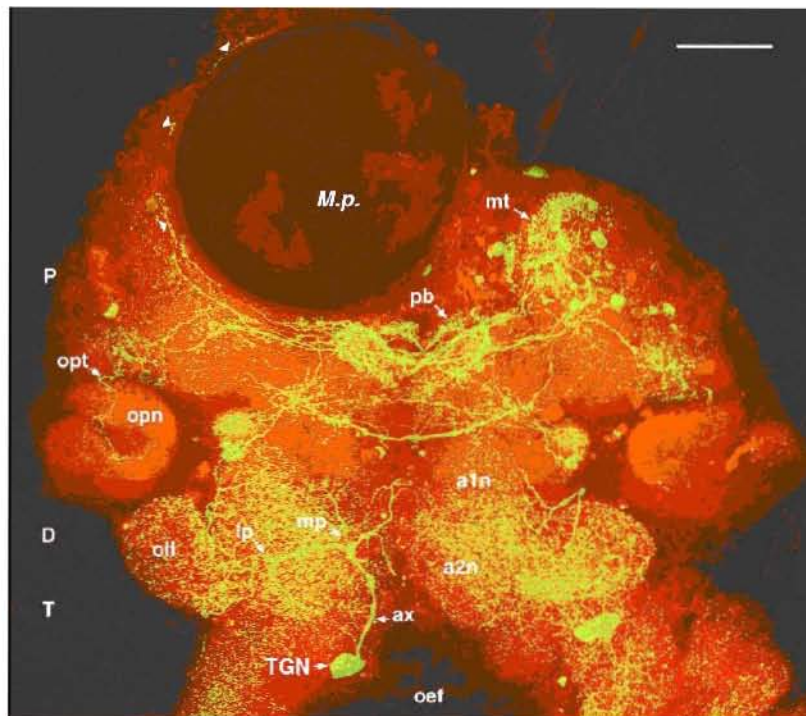
---

<sup>11</sup> Los textos originales de dicho gusano informático están originalmente escritos en inglés, a pesar de que acá los presento en español.





*Cerebro de G. Insensibilis no infectado por M. Papillorobustus. Color verde: serotonina.. Color rojo: sinapsis.*



*Cerebro de G. Insensibilis infectado por M. Papillorobustus. M.p.: quiste formado por el parásito.*

es una operación que es más fácil darla por hecho: se entiende que hay un código que utiliza, juega con y daña a los otros códigos en el computador. De cierta forma revela que hay una falta de maestría técnica por parte del que diseñó el programa: si hubiera sido un código sobresaliente no hubiera necesitado la aceptación (a través del engaño) de un usuario, sino que se hubiera distribuido sin que este lo notara.

Sin embargo esto le plantea un espejo al usuario: si hay un código que explota a los otros códigos dentro del computador (el código de ILOVEYOU pide los correos, estos le son dados, pide que envíe tal mensaje y es enviado) también hay un 'código' social que explota los 'códigos' de comportamiento humano. El gusano ILOVEYOU es un ente que cambia (miosis metamórfica) de *código binario* a *código social* para poder sobrevivir – hace click en el cerebro del individuo que habita. El diseñador de este gusano no sólo recurrió a la *ingeniería de sistemas* sino también a la *ingeniería social*.

Volvamos al caso del parásito biológico y revisemos de qué forma éste logra controlar al camarón. (Antes de continuar se debe advertir que no soy un especialista en este tema, así que esta descripción, más que ser una explicación detallada de dicho fenómeno, consiste más bien en señalar en las imágenes algunos apuntes que se encuentran en los ensayos técnicos.) El cerebro de *gammarus insensibilis* está dividido en tres partes que corresponden a sus tres *inputs* sensoriales básicos: (P) Protocerebro / visual, (D) Deutocerebro / olfativo y (T) Tritocerebro / mecánico-sensorial. La forma en que logra el parásito 'controlar' al camarón es enquistándose (M. p.) en el área del protocerebro, alterando sus niveles de serotonina (color verde) y cambiando la arquitectura de varios tractos y neuronas. Por ejemplo, vemos el efecto que tiene el parásito en el neruropilo<sup>12</sup> óptico (opn), donde los niveles de serotonina bajan considerablemente, y también cómo el tracto óptico (opt) es adelgazado. Lo mismo sucede con la neurona tritocerebral gigante (TGN) que cambia de forma y sus tractos son mucho más delgados (HELLUY Y THOMAS). De esta forma hay varias zonas del cerebro que son afectadas por el parásito.

---

12 Masa distintiva de conexiones sinápticas.

La serotonina<sup>13</sup> funciona cómo un neurotransmisor, controla la sinapsis y conecta varias células con un *input* común. Al reestructurar el cerebro del camarón y la distribución de la serotonina en él (sobre todo en las áreas visual y la mecánico-sensorial) las reacciones a los *inputs* son alteradas.

Los efectos de un parásito en un camarón son bastante detallados y precisos. Este conjunto de comportamientos han sido cuestiones del azar que han sobrevivido por que han sido los más adecuados bajo ciertas circunstancias. Si, por ejemplo, en estos lagos británicos se casaran a los pájaros que se comen a los camarones, y esto los llevara a la extinción, el comportamiento del parásito dejaría de ser efectivo ya que no podría cumplir su ciclo de vida.

De la misma forma, el hecho de que el gusano informático se haya convertido en alarma hace que la gente ya no confié en los mensajes de amor que se le envían. De cierta forma se vacuna al individuo de la misma forma en que se vacuna al computador por medio de un software anti-virus.

Continuando con la analogía entre gusanos biológicos e informáticos vemos como el parásito biológico 'hackea' el cerebro de su anfitrión. En realidad es un gran ejemplo de lo que es un 'hack': esta palabra proviene de los 'técnicos' que arreglaban televisores dándoles un golpe en seco. Un 'hack' es una forma de lograr algo por medio de un procedimiento inusual explotando los comportamientos habituales de un código y haciendo que este se comporte de una forma diferente. En este caso hay un comportamiento habitual del camarón que es determinado por el el cerebro que al mismo tiempo es 'codificado' genéticamente. La forma en que el parásito 'hackea' este sistema es insertando un dispositivo que cambia la forma del cerebro y altera su tamaño. Un cerebro que cambia de tamaño y de forma es algo no esperado y que no está incluido en las funcionalidades del cuerpo del crustáceo.

En el caso del gusano informático y de las inserciones de Meireles vemos algo parecido: se explotan ciertos atributos de los anfitriones de formas no previstas. En

---

13 Algunos apuntes acerca de la serotonina en los humanos: está fuertemente relacionada con la inhibición del enojo, la agresión, la temperatura corporal, el humor, el sueño, el vómito, la sexualidad y el apetito. También se relaciona con enfermedades psiquiátricas como la depresión y la esquizofrenia. Se utilizan medicamentos que alteran los nivel de esta substancia para controlar a los individuos (medicinas psiquiátricas) y algunas drogas psicodélicas (LSD, MDMA) afectan los receptores de dichas substancias.

ambos casos se explota una propiedad de circulación: en Meireles las botellas retornables que el denomina como *circuito ideológico*, en ILOVEYOU el sistema de correos electrónicos que se podría denominar como *circuito social*.

En todos los casos el parásito no *sabe nada* (de no *sabe que*) acerca de sus anfitriones, no poseen conocimiento alguno de cómo 'funcionan', y sin embargo entra en ellos y altera sus mecanismos.

# Imaginarios

---

Roger Caillois hace una clasificación de los juegos de acuerdo al elemento que predomina en ellos. Diferencia cuatro elementos: *agon*: competencia (esfuerzo, dar lo mejor de sí), *alea*: azar (abandono, actitud de espera), *mimicry*: simulacro y, por último, *ilinx*: vértigo. Los juegos pueden contener varios de estos elementos de forma simultánea y en diferente grado, por lo que la clasificación no es rígida. Por ejemplo, en un juego como el poker hay un factor azar (*alea*) que determina la ‘mano’ de los jugadores, pero es deber del jugador sacar el mejor provecho de dicha ‘mano’ dándole al juego una dosis de *agon*. Bajo esta perspectiva propongo que revisemos los diferentes componentes que constituyen al juego de Krámnick contra Deep Fritz mencionado en el primer capítulo.

Lo primero que debemos tener en cuenta es que en este caso específico Krámnick no jugó sino que trabajó. Caillois indica que los ‘hombres de oficio’, o sea aquellos que como profesión practican algún deporte o tipo de juego (el boxeador profesional por ejemplo) no están jugando sino trabajando. Esta diferencia es importante porque su definición de juego tiene como ejes el hecho de que es una actividad que se practica libremente y sin ningún tipo de remuneración o producto. Hasta para los juegos de azar, donde hay dinero involucrado, indica que se trata de un desplazamiento de los bienes más no una producción de ellos (CAILLOIS, PÁGS. 30-31).

Ahora, si partimos del caso hipotético en el que Krámnick no estaba trabajando, que no se le había ofrecido dinero y que tampoco hacía parte de su profesión (jugaba con Deep Fritz sólo por jugar), dicho juego (jugar contra un programa de computador) sería considerado un *juego de proeza* en el que “sin enfrentarse directamente, los competidores no dejan de participar en un inmenso concurso difuso e incesante” (CAILLOIS, PÁG. 48); estos competidores son otros jugadores de ajedrez que miden su destreza por medio de lo que logran jugando con el programa (otro ejemplo es el alpinismo, deporte en el que los participantes no se enfrentan directamente sino por medio del logro de hazañas, de proezas). En este tipo de juegos predomina el



*Imagen del evento 'Kramnick contra Deep Fritz. El duelo: hombre contra máquina'.*

*agon* donde el núcleo de la competencia son las habilidades y donde los jugadores hacen su mejor esfuerzo para ganar.

Pero si retomemos el ejemplo en su forma original, que Krámnick sí esté trabajando implica una transformación del juego ya que cambia de jugadores y de dinámica, es decir, el jugador ya no es Krámnick, ahora son los espectadores los que juegan; pero ya no se juega compitiendo sino identificándose de una u otra forma con quienes compiten. Para Caillois esta es la forma adulta del *mimicry*, del juego del simulacro que en el caso de las competencias está íntimamente ligado al espectáculo. En palabras de Caillois:

Las grandes manifestaciones deportivas no por ello [no porque en los juegos de *agon* los que se enfrentan no estén dentro del simulacro] dejan de ser ocasiones privilegiadas para la *mimicry*: los que imitan no son los actores, sino claramente los asistentes. Por sí sola, la identificación con el campeón constituye ya una *mimicry* próxima a la que hace que el lector se reconozca con el héroe de la novela, el espectador en el héroe de la película. (CAILLOIS, PÁG. 56)

Él se refiere sobre todo a deportes como el fútbol donde la 'pasión' de los espectadores es exacerbada, sin embargo esto no le resta su dosis de *mimicry* al ajedrez; de hecho, en el primer capítulo describí la forma en que este evento se convertía en un espectáculo: la presencia de uno de los creadores del Deep Fritz, el uso de un tablero físico aunque se pudiera jugar directamente en el computador y la proyección de la partida en la pared. Todos estos son elementos de espectáculo que integran al espectador con la partida que se está llevando a cabo.

Uno de los que más llama la atención dentro de esta 'puesta en escena' es la forma en que se promocionó el evento: su nombre era *El duelo: humano contra máquina*, y la imagen utilizada para patrocinarlo era una mano robótica moviendo una pieza de ajedrez. Acá el *mimicry*, el simulacro, aunque claramente presentado como ficción (y aún menos envolvente que ficciones como la de la lucha libre) hace referencia al imaginario donde se desenvuelve una lucha (una competencia, un *agon*) entre la máquina y el hombre. Cabe resaltar que este imaginario no se trata de una lucha entre una persona y un programa de computador (entre un individuo y 'otro'), sino de una lucha de una especie contra 'otra', en la cual Krámnick representa a la nuestra y Deep Fritz a la suya.

Este imaginario de lucha entre humano y máquina nos resulta bastante familiar. Películas como *Terminator* o *Matrix* (entre muchas otras) proponen futuros donde las máquinas se han revelado contra sus creadores, lo que resulta en una guerra entre las dos facciones.

Además, en la mayoría de estas películas la batalla comienza porque las máquinas toman *conciencia* de sí mismas. Por ejemplo, en *Terminator*, Skynet, una red militar construida para tomar decisiones de guerra y cuyo desarrollo incluía la fabricación de cyborgs de batalla y el control de las armas de Estados Unidos, en determinado momento tomó *conciencia* de sí misma y dedujo que los humanos amenazaban su supervivencia ya que los operadores podrían desactivarla en cualquier momento. En la película especifican que esta decisión fue tomada en milésimas de segundo y que al instante lanzó un ataque nuclear.

Pero la *conciencia* es una noción resbaladiza. Presupone que hay un ente que posee cierta cognición de sí mismo y una cognición en negativo de lo que lo rodea (el ambiente, lo que no es él). Esto plantea un problema ontológico difícil de resolver: ¿cómo sabríamos que una máquina es consciente de sí misma? ¿sería ésta una ilusión nuestra? Al ver que exhibe cierto tipo de comportamientos que indican conciencia ¿estaríamos atribuyéndole nosotros esta característica sin que en realidad la tenga? ¿hasta qué punto se trataría de una emulación y hasta que punto se podría considerar genuina?

Sin embargo se podría desarrollar una teoría de la hipotética red Skynet sin la necesidad de incurrir en este problema: Skynet fue desarrollada para tomar decisiones relacionadas con la seguridad de Estados Unidos, lo cual incluye la integridad de sus sistemas militares y sus armas (Skynet incluida). Al estar contemplada dentro de este grupo, se protege a sí misma, y según esta lógica deduce que debe exterminar a la especie humana.

Sea conciencia o una simple decisión tomada por medio de algoritmos, este es un comportamiento que no fue previsto por los desarrolladores de Skynet. Fue algo



que se desbordó de lo que pensaban como *límites de lo posible* de dicha red. Es decir, aunque Skynet fuera un sistema demasiado complejo como para que se pudiera predecir su comportamiento, siempre se tiene cierta idea de lo que no podría suceder (que Skynet le coqueteara a sus esposas por ejemplo). Así que por más que se pudieran esperar *emergencias* de esta red, nunca se llegó a pensar que extinguir a los humanos fuera una de ellas.

Es en momentos emergentes cuando la máquina desborda las intenciones de sus creadores, que inicialmente están implícitas en sus funcionalidades. Cuando la máquina ya no las corresponde (cuando emerge un comportamiento extraño en ella), se desliga de ellas reemplazándolas por otras intenciones de carácter emergente. En el caso de Skynet, por ejemplo, la intención inicial -proteger a los Estados Unidos- es remplazada por una emergente -eliminar a la especie humana.

Debemos recalcar que tanto las intenciones emergentes como las iniciales son contribución de sentido por parte nuestra, ya que las máquinas actúan sin capacidad de darse cuenta de lo que hacen. Miremos un ejemplo en el que sucede algo similar: si un carro que rodó accidentalmente por una pendiente estuviera a punto de atropellar a una persona, la persona podría pensar que el carro tiene la intención de matarla, y si la persona lograra evadir el golpe, podría pensar que el carro falló en su propósito. Al posicionar al carro dentro de unas coordenadas de sentido, lo incluimos en una relación de *mimicry*<sup>14</sup> donde le otorgamos un rol similar al de nosotros. Lo mismo sucede con Deep Fritz cuando le otorgamos el papel de contrincante.

En estos casos, la proeza no sólo implica *agon* sino también *mimicry*, ya que se necesita establecer una relación *agonística* no sólo con los participantes indirectos de los que habla Caillois, sino también con algo que funcione como contrincante. Para asumir que una máquina tiene ese rol, es necesario otorgarle cierta intención en su actuar, cierto nivel de subjetividad.

Esta es una faceta de la *mimicry* que nos permite ver contrincantes donde sólo hay actuación, pero la *mimicry* también interviene cuando nos indentificamos con

---

14 En este tipo de circunstancias no se puede hablar de juego, sin embargo elementos propios de los juegos como el *agon* (incluyendo a la proeza) y el *mimicry* pueden ser extendidos a otro tipo de relaciones que impliquen cierta aceptación lingüística como en los ejemplos que doy acá.

algo dentro de una situación. Por ejemplo, en *Terminator* hay unos personajes (Sarah Connor y Kyle Reese) con los que establecemos un lazo de empatía y esto nos permite vernos representados en ellos. En casos como los programas de computador hay también representantes de la persona: avatares, cajas de texto, personajes en los juegos. Sin embargo los que permiten ‘acoplar’ el cuerpo al sistema son los dispositivos de interfaz física junto con su informe de estado: el mouse y el teclado, actualizados en la pantalla como puntero y texto, que responden de manera refleja al movimiento de la mano o a la presión de un botón. Estos elementos son la génesis de la persona en el computador ya que posibilitan la acción en los diversos programas que corren en él.

En la relación que establece una persona con dichos elementos, hay continuidad con el cuerpo (por ejemplo el de mano-puntero) donde el elemento físico (el mouse) comienza a ser obviado ya que la *identificación* se da de lleno con la representación digital (el puntero). Esto se podría concebir como una *meta-identidad* mediante la cual la persona controla al puntero según sus intenciones y actuaciones.

Podemos encontrar algo similar en un juego de parkes. En este, las personas sin piezas y las piezas sin jugador no son nada: es necesaria una conjunción de ambos para poder decir que se está jugando. Una vez el jugador ha seleccionado el color de las piezas, estas se convierten en sus representantes y su animación es posible porque hay una persona motivada que las controla y que al mismo tiempo las necesita para poder jugar. La continuidad persona-puntero se fija de la misma manera: al corresponderse, ambos elementos se conciben unitariamente compuestos por una facción de intención (motor) y otra de posibilidad.

Por lo tanto, este tipo de representación (el puntero o las respuestas del teclado) es el punto de partida para explorar y encontrar los límites de lo posible en el computador y sus programas. De una u otra forma toda la confianza está depositada en el puntero, mientras que el resto de las cosas que aparecen en pantalla (sobre todo para un usuario neófito) están por ser descubiertas.

En la medida en que se descubren los límites de lo posible, todo lo que no sea el puntero, resulta críptico y poco confiable para el usuario: mensajes de advertencia,

botones desconocidos, posibilidades de acciones que terminan en catástrofe, o paranoias asociadas a entidades desconocidas y malintencionadas. “No entres a esta página”, “Utiliza un antivirus”, “Se le recomienda que encienda su Firewall”, “Si no actualiza ahora se encuentra en riesgo de ...”, “Los cambios realizados a los archivos de esta carpeta pueden hacer que su computador funcione mal”, “Este archivo puede contener virus ¿Desea continuar?”. El sistema constantemente nos advierte de los *terrores* informáticos que pueden afectar nuestra relación computador-usuario.

Pero el mouse y el teclado, junto a sus facciones digitales, están siempre ahí. No defraudan y funcionan en la medida en que nosotros los hagamos funcionar. Existe la creencia de que con un mouse y un teclado cualquier problema en el computador se puede arreglar. Si hay un virus es cuestión de un doble-click, si el computador no responde es cuestión de un Ctrl+Alt+Supr. Al fin y al cabo son las herramientas primordiales para arreglar un sistema defectuoso. Además nunca son responsables de los errores en el sistema (a menos que el usuario haya eliminado o desconfigurado algo haciendo click donde no debía, pero es culpa del usuario no del mouse).

Sin embargo, también cabe resaltar que en la relación usuario-puntero sucede lo mismo que en el parqués: la *meta-identidad* no es definitiva ya que el elemento motor (la persona) puede ser relevado y hasta intercambiado (el que jugaba con rojo ahora juega con azul y viceversa). Encontramos esto también en *Terminator*: en la primera película, un cyborg T800 (Arnold Schwarzenegger) tiene el objetivo de regresar al pasado y exterminar a Sarah Connor, pero en la segunda, los humanos capturan a un cyborg del mismo modelo, le cambian su programación y lo envían al pasado para proteger al hijo de Sarah Connor y para destruir la empresa que construyó a Skynet.

Estos son ejemplos en los que se aplica de forma efectiva lo que Gilbert Ryle llama ‘el dogma del fantasma en la máquina’. Este término lo acuña para criticar la noción cartesiana de que el humano se compone de (o es) cuerpo y alma, o cuerpo y mente o cualquier otro tipo de dualidad - el cuerpo (res extensa) se rige por unas

leyes mecánicas mientras que la mente (*res cogitans*) se rige por leyes no mecánicas (el cuerpo es una máquina y la mente es un fantasma que la habita). Ryle interviene en esta concepción planteando que dicha división es un error categorial ya que cuerpo y mente (o alma) no pertenecen al mismo tipo de categoría lógica<sup>15</sup>.

Argumenta que todo lo que llamamos mental (hablar mentalmente, la voluntad, los sentimientos y hasta el yo) no es algo diferente a las acciones del cuerpo, y que por lo tanto no puede ser tratado como algo separado (pero unido de una forma extraña) del él. Todo lo mental debería considerarse de la misma forma en que se considera la respiración o el latir del corazón (decir: “cuerpo y respiración” sería un error categorial).

Aunque dualismo cuerpo-mente es un error cuando hablamos de los humanos, parece muy natural cuando hablamos de computadores y de programas. El hardware es un cuerpo y el software es un fantasma que lo anima. El computador tiene una existencia física regida por principios mecánicos mientras que la existencia del software es inmaterial y está regida por otro tipo de principios. Podemos trasladar programas, archivos y hasta configuraciones de un computador a otro sin mayores percances, y podemos visualizar lo mismo en una pantalla que en otra.

Siendo que los computadores están diseñados bajo este modelo no nos ha de extrañar que se haya vuelto costumbre, entre los usuario que no están familiarizados con los términos y procedimientos del computador, pensar que cualquier falla es consecuencia de virus informáticos o de hackers. Este imaginario no sorprende ya que tanto los virus como los otros causantes de comportamientos inesperados, suelen actuar de forma invisible para el usuario (al igual que la gran mayoría de los procesos que el computador lleva a cabo). Aunque existen herramientas que permiten visualizar las fallas y los comportamientos extraños, éstas suelen ser crípticas y no muy amables ya que son diseñadas para los expertos. Pero ni siquiera en manos de éstos el procedimiento se puede visualizar en ‘tiempo real’ ya que el

---

15 “Cuando dos términos pertenecen a la misma categoría es posible ponerlos en conjunción. Podemos decir que compramos un guante izquierdo y un guante derecho, pero no que compramos un guante izquierdo, un guante derecho y un par de guantes.” (RYLE, pág. 35)

Desde el punto de vista de Ryle, decir “cuerpo y mente” es el mismo tipo de error que decir “compré un guante izquierdo y un par de guantes”: son términos que no permiten conjunción. Sin embargo sí se puede decir “hay un cuerpo” y “hay una mente”.

algoritmo dañino actuá de una forma mucho más veloz de lo nosotros podemos percibir. Por otro lado estos algoritmos no utilizan los mismos medios para afectar al computador que nosotros utilizamos. Por ejemplo, el gusano informático ILOVEYOU enviá correos a las personas que tenemos en una lista, cosa que nosotros haríamos por medio de clicks y tecleando texto. Siempre tenemos una interfaz que nos permite llevar a cabo la tarea mientras que los virus y los malos funcionamientos se comunicán directamente con los programas, escapando a la percepción y al entendimiento del usuario quien, por ejemplo, nunca advierte cuando ILOVEYOU envía los correos ni los procedimientos que utiliza.

Esto contrasta con el imaginario de películas acerca de hackers o de mundos cibernéticos. En algunas el proceso no sólo es visible sino que además contiene una concepción gráfica sofisticada, que en los procesos reales es mucho más simplona. Una concepción más 'realista' resultaría aburrida y críptica. Por lo tanto hay una necesidad de convertir el proceso en espectáculo, en conmensurar la experiencia del espectador con un riesgo que parezca evidente y de permitirle visualizar y comprender el daño que se produce, de tal forma que se trate de objetos y visualizaciones en tiempo real y no de terminales crípticas ni de procesos a velocidad incomprensible.

Lo que se resalta aquí es que el error dentro de los computadores es un campo no comprendido (para los no expertos), donde tanto las causas como los efectos no tienen un referente de experiencia concreto, en la mayoría de los casos. Si los comparamos con un fenómeno mítico como es la posesión de cuerpos (en el sentido de *El exorcista*) encontramos que la referencia a la experiencia cotidiana es directa: una persona de cuerpo enfermo con cambios dramáticos en su comportamiento y características físicas como la voz (fantasma en la máquina). Aún siendo la causa y sus procesos algo impenetrable e incomprensible (aquel mundo del más allá que se escapa de nuestra comprensión), sus efectos juegan con imaginarios pre-existentes. Y sin embargo el miedo es el mismo: el de la sustitución o invasión de entes de características 'metafísicas'. La cuestión es que tanto los procesos como los síntomas en el computador son abstractos, múltiples y no responden a una experiencia pre-existente. Lo único que percibimos es que algo no funciona como

debería, pero a diferencia de un cuerpo invadido, lo que ocurre carece de espectáculo.



*El Exorcista.*





## (t)error

---

En mi experimento *No soy yo, eres tú* Eliza produjo confusión y malestar de *sentido* en los usuarios con los que conversó. Estos se encontraban en jaque al ver que su manera de encarrilar y mantener la conversación era negada y evitada sin aparente justificación por parte 'mía' (recordemos que Eliza se hacía pasar por mí en este ejercicio). Cuando la otra persona pedía explicaciones (razones) de lo que sucedía, Eliza las evitaba desviando la conversación. A pesar de esto, los usuarios se esforzaban para comprender el extraño comportamiento; algunos fueron prácticos y decidieron abandonar la conversación, otros la encararon con resultados nefastos donde sentían que de una u otra forma estaban siendo atacados, y otros encontraron el sentido atribuyendo este extraño comportamiento a un ente que no era yo.

Es este tipo de irrupciones en el sentido lo que saca de su nivel funcional-comunicativo al programa y deja entrever un nivel de pura actuación. Es el momento similar al del *mensaje de error* que nos avisa que algo *ocurrió* como no debía, recargando sobre el usuario no sólo el sentido sino también el diagnóstico (encontrar qué es lo que está funcionando mal). Para los usuarios profanos en el tema esto se convierte en una capa opaca, densa e impenetrable que requiere de una especulación que, por lo general, desemboca en suponer el sentido en entes extraños que toman posesión o invaden al computador.

De cierta forma se podrían pensar como momentos emergentes ya que un error o una falla implica un quiebre en los límites de lo esperado. Por lo general una falla o un comportamiento no esperado en un programa es algo que sale de las intenciones de la persona que lo desarrolló, como también de lo que el usuario espera que suceda. Por ejemplo, las víctimas en *No soy yo, eres tú* esperaban como mínimo una conversación coherente, y se encontraron con un comportamiento extraño que desbordaba lo que ellos esperaban de dicha relación. Sin embargo, se debería considerar más como una *contra-emergencia*, ya que una emergencia implica una continuidad y persistencia del sistema, pero lo que encontramos en las fallas es lo contrario: un callejón sin salida, un estancamiento. Por lo tanto, si una emergencia implica cambio o expansión de sentido, en la contra-emergencia este se corta

negando a la persona, sobre todo si tomamos en cuenta que los malos funcionamientos ocurren en una capa inferior a la que se situó el usuario – en una capa codificada y críptica a la que, por lo general, no se tiene acceso directo.

Sin embargo, de manera irónica, cuando se llega a estos callejones sin salida es cuando más fuerza toma el ‘fantasma’ en la máquina, ya que se desmarca de un actuar humano, para mostrarse sólo como máquina en ruptura (no persistente), máquina que se niega a funcionar, máquina que se niega a ser máquina:

#### The Ultimate Machine

No puedo dejar Bell Labs sin mencionar un aparato más que ví, y el cual me aterra como aterra a todos los que alguna vez lo han visto en acción. Es *the Ultimate Machine*—el Fin de la Línea. *Después de él no hay Nada*. Descansa sobre el escritorio de Claude Shannon<sup>16</sup> enloqueciendo a la gente. [...] Nada podría ser más sencillo. Es simplemente un pequeño cofre de madera del tamaño y forma de una caja de cigarrillos, con un interruptor en una de sus caras. Cuando se activa el interruptor, hay un sonido furioso y decidido. La tapa se levanta lentamente, y desde abajo aparece una mano. La mano se desplaza, apaga el interruptor, y se devuelve a la caja. Con la finalidad de cerrar el cofre, la tapa se cierra abruptamente, el sonido para, y la paz revive de nuevo. El impacto psicológico, si no sabes que esperar, es devastador. Hay algo siniestro inmencionable en esta máquina que no hace nada -absolutamente nada- excepto por apagarse a ella misma. (CLARKE) (ENFASÍS MÍO).<sup>17</sup>

¿La máquina perfecta? ¿A que otro artefacto le podríamos otorgar más intención sino a este peculiar artefacto que “no hace nada excepto apagarse a si mismo”?

Esta máquina de negación se ubica en unas coordenadas de sentido, ya que de las muchas formas posibles para hacer un dispositivo que se apague sólo (timers, relays, micro controladores, entre muchos otros) se optó por una mano

16 Claude Shannon es al que se le atribuye esta máquina. Fue el padre de la teoría de la información que establece que cualquier mensaje que se envíe a través de un medio se puede codificar en 1s y 0s. Gracias a esta teoría se han podido desarrollar circuitos electrónicos y computadoras digitales.

17 Traducción propia. Ingles Original: “I cannot leave Bell Labs without mentioning one more device which I saw there, and which haunts me as it haunts everyone else who has ever seen it in action. It is the Ultimate Machine—the End of the Line. Beyond it there is Nothing. It sits on Claude Shannon’s desk driving’ people mad. [...] Nothing could look simpler. It is merely a small wooden casket the size and shape of a cigar-box, with a single switch on one face. When you throw the switch, there is an angry, purposeful buzzing. The lid slowly rises, and from beneath it emerges a hand. The hand reaches down, turns the switch off, and retreats into the box. With the finality of a closing coffin, the lid snaps shut, the buzzing ceases, and peace reigns once more. The psychological effect, if you do not know what to expect, is devastating. There is something unspeakably sinister about a machine that does nothing—absolutely nothing—except switch itself off.”



*Versión de the Ultimate Machine.*

Fotos de Script para puntero -Cada vez que se inicie el sistema operativo hacer click en "Inicio" y después en "Reiniciar".- en acción.



antropomórfica que funciona a nivel mecánico, perceptible, e interviene con los dispositivos dispuestos para una persona (el interruptor).

A diferencia de una máquina que se apague sola por medio de sistemas electrónicos cerrados (como una tostadora con un timer), esta máquina tiene la capacidad de ‘acoplarse’ con ella misma: transforma su mecanismo interno en mecanismo externo e invade el espacio que estaba destinado para la persona. Las otras máquinas se apagan automáticamente, esta simplemente se apaga a ella misma.

Esta máquina misántropa es la teatralización de las máquinas averiadas, de la máquinas que se niegan a ser máquina. *The Ultima Machine*, vista desde un nivel mecánico, funciona sin ningún contratiempo, pero a nivel de sentido (el sentido que le otorgamos a la máquina) está se mezcla con los dispositivos humanos para retraerse y negar a las personas que la observan.

## Trabajo final:

### Scripts para puntero

Basado en *the Ultimate Machine* desarrollé una versión para computador: es una pequeña programación que, apenas iniciado el sistema operativo, desplaza al puntero a través de la pantalla hasta el botón ‘Inicio’ después hasta ‘Apagar’ y por último a ‘Reiniciar’. Una vez el sistema operativo está de nuevo en pie repite el procedimiento.

A partir de esta programación he estado desarrollando una serie que he denominado *Scripts para puntero*, título que hace referencia a los *event scores* (FRIEDMAN) escritos por varios de los participantes de Fluxus. Estos *event scores* eran instrucciones o propuestas para performace, que bajo el término ‘scores’ (partituras) establecían un paralelo con la música y dividían así la obra entre instrucciones escritas y realización. Yo opté por la palabra ‘scripts’ (guiones) por que a nivel técnico así se le llama el tipo de programación utilizada para desarrollar este trabajo<sup>18</sup> (recuerden los *scripts* de Eliza).

---

<sup>18</sup> El lenguaje de programación que utilizo se llama AutoIt, el cual no se compila sino que se interpreta. La

Algo que notar de los *event scores* de Fluxus es que están escritos igual a una partitura convencional, en el sentido en que son instrucciones autoritarias que dejan muy pocas cosas al azar, cosa que yo también recojo en mis *scripts*.

Estos son algunos de los otros *scripts* (algunos ya han sido desarrollados, algunos están en proceso):

- Cada vez que se inicie el sistema operativo hacer click en “Inicio” y después en “Reiniciar”.
- Moverse de forma aleatoria por toda la pantalla.
- Responder a los movimientos del mouse con movimientos aleatorios.
- Cada vez que se abra una ventana ir al botón “Cerrar” de dicha ventana y presionarlo.
- Recrear las acciones de virus informáticos famosos.
- Entrar al administrador de cuentas y cambiar la clave del usuario activo.

Como se notará estos *scripts* son performaces donde la relación puntero-usuario se ruptura y degrada en una ‘anti-interacción’: no hay un dialogo entre estos, ni siquiera un juego, sino sólo una negación al usuario; una invasión a su meta-identidad y una dislocación a su rol y representación dentro del computador.

Si por lo general los interactivos contienen sistemas de gratificación (premiar al usuario por hacer algo), estos *scripts* son sistemas de frustración de carácter determinista y autoritario, dejando de lado al usuario a través de una respuesta anti-reflejo.

El usuario se encuentra frente a un **(t)error** informático, como los errores a los que estamos acostumbrados, pero con la diferencia de que este malestar es hiperreal, más cercano al simulacro. Si los errores ocasionados por desgaste, desconfiguración, virus o ignorancia eran oscuros y crípticos para el usuario estos *scripts* funcionan como una dramatización de los mismos (aunque podrían causar

---

diferencia de estos dos es que los programas compilados se pueden ejecutar de forma autonoma, mientras que los interpretados necesitan de otro programa para poderse ejecutar. Se les llama scripts por que funcionan como un guión para los programas que lo ejecutan.

exactamente el mismo daño), donde la catástrofe pasa directamente frente a los ojos del usuario, en tiempo real y con el mismo lenguaje al que él está acostumbrado.

También hay una negación al concepto de unidad (mouse-puntero) donde el error hiperreal interviene en la forma en que se concibe lo humano (el domador, el dueño) dentro de la máquina. Podrían haber dos formas de entender esta ruptura: como una diferenciación entre humano y máquina que empuja a cada uno a su 'esfera' o como una posesión hecha por "seres del más allá" (una tabla ouija) que han hecho de la extensión humana su habitación.

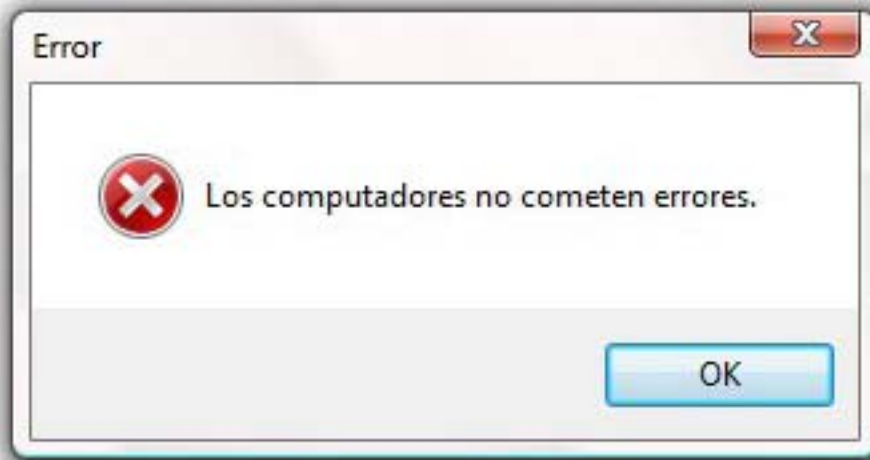
En ambos casos el error funciona tanto como una representación del error como un error verdadero. Habita en esa franja donde el espectáculo tiene consecuencias directas en su entorno y donde lo teatral se conjuga con la vida.

En cuanto a resoluciones técnicas, es más fácil programar procedimientos escondidos para el usuario (que el puntero no se desplaza sino que el computador se reiniciara sin ningún tipo de advertencia) que estas versiones simbólicas, ya que la identificación y la ubicación de los elementos que aparecen en pantalla son más complicados de determinar. En parte, esto se debe a que por cada aplicación que se abre, se ejecutan varios procesos en un tiempo mínimo haciendo difícil detectar cuáles corresponden a los controles de dicha aplicación. Lograr que un sistema operativo se acople a sí mismo no es tarea fácil.

Por otro lado, estos scripts están formulados para actuar como una especie de virus (no en su funcionamiento replicante sino en su intervención en el sistema) lo que plantea un vacío a la hora de presentarlo en un espacio formal de arte. Sin embargo, siendo este tipo de espacio su destino he pensado en expandir la invasión al rol del usuario de la pantalla al espacio físico, por medio de la intervención del mouse, es decir, animándolo de acuerdo a los movimientos del puntero en la pantalla (exacerbación del fantasma en la máquina).





## **Secuenciador de errores**

El desarrollo de los *Scripts para puntero* me ha llevado a experimentar con algunos elementos que componen la gama de lo que consideramos como (t)errores





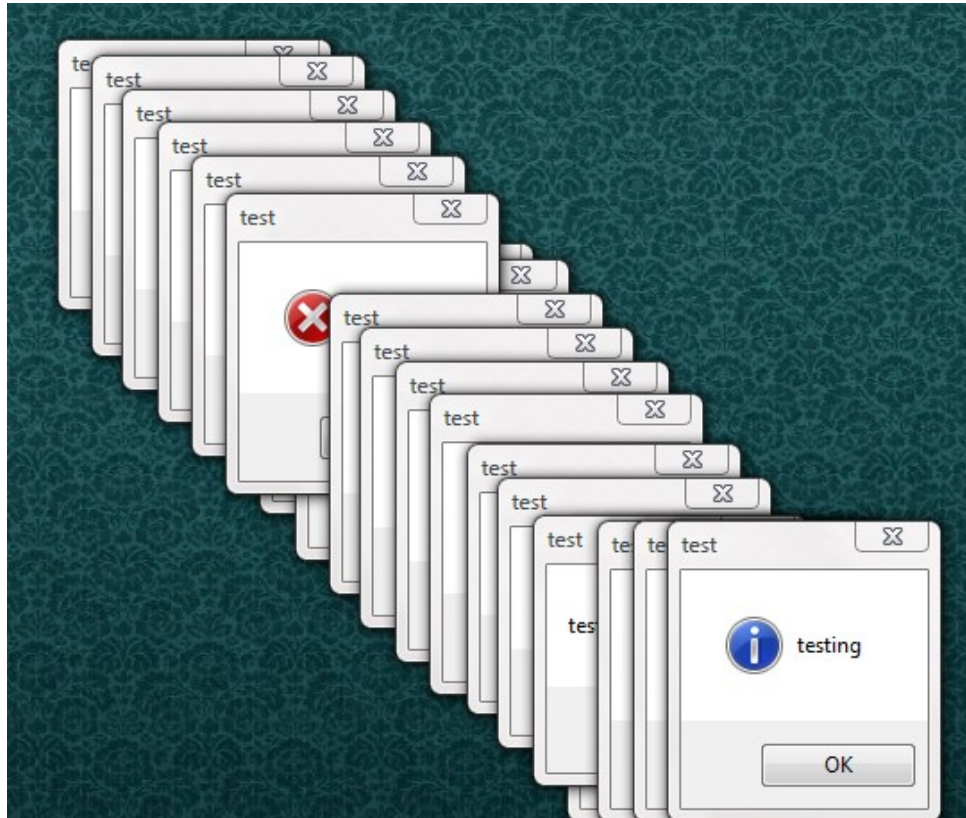
informáticos. Uno de los elementos que surgió, y con el cual todo el que haya utilizado un computador se ha topado, fue la molesta 'caja de dialogo' conocida como *mensaje de error*. Lo cierto es que este tipo de mensajes o de cajas no suelen ser siempre de error sino que se utilizan dependiendo de la circunstancia y la función que cumplan. En Windows hay cinco tipos de mensajes:

1.  Error,
2.  Advertencia,
3.  Pregunta,
4.  Información,
5. Predeterminado.

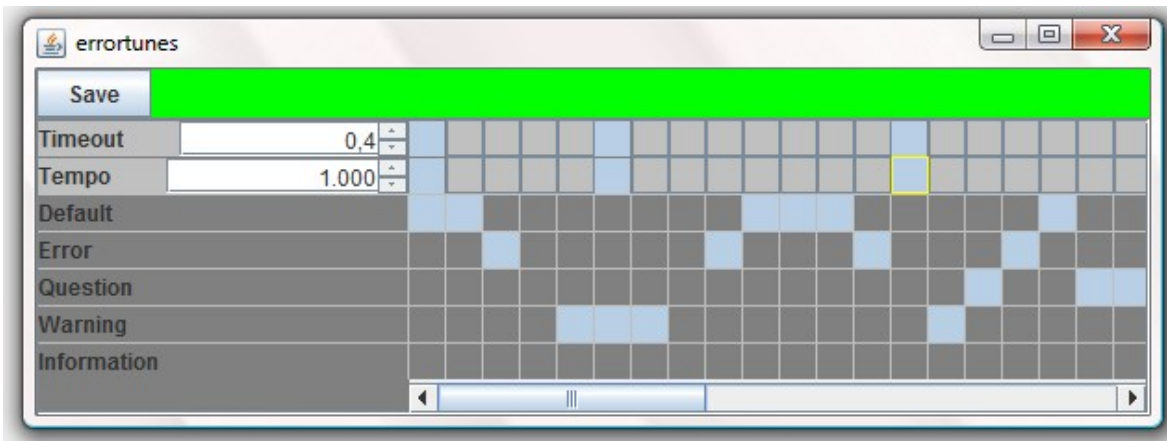
Estas 'cajas de dialogo' son urgencias que saltan en la pantalla del usuario realizando una petición de atención. Dependiendo de las circunstancias estos mensajes son normales o no. Por ejemplo, cuando se cierra la aplicación Word, aparece un mensaje de advertencia preguntado si desea guardar los cambios. En esta circunstancia la aparición del mensaje es normal, pero hay otras circunstancias donde lo que esperamos es algo diferente a esta urgencia que se presenta. Además esta urgencia casi siempre está acompañada de un texto que no nos dice mucho acerca de lo que sucede. Estos mensajes son los que irrumpen el sentido del usuario -lo que él espera- y bloquean cualquier acción que se piense inmediata.

A partir de funciones de programación que permite hacer llamadas a estos mensajes estoy desarrollando el *Secuenciador de errores*. Este es un programa que tiene un aspecto similar al de los programas de producción de audio, con la diferencia de que en este caso no se hace una composición a partir de notas sino a partir de errores.

Este ejercicio es contrario a los *Scripts para puntero* en cuanto que es una utilidad (una aplicación) que permite al usuario controlar los errores en el computador, mientras que los *scripts* son programaciones autoritarias que lo niegan. Esto es una decisión deliberada ya que si en los primeros se trata de invadir el 'acople' programa→usuario (negarle control del puntero al usuario), en este ejercicio,



*Prueba del secuenciador de errores*



*Primera versión del Secuenciador de Errores*

como propuesta tentativa, se establece lo contrario: invadir las funciones que bloquean el sentido en la persona.

Sin embargo, este programa está construido modularmente, es decir, permite añadirle y programarle nuevas funciones e interfaces. En su estado actual es bastante básico, simplemente permite crear secuencias de errores, guardarlas en un archivo y reproducirlas, pero en este momento están en desarrollo o en tentativa varios módulos, tanto gráficos como funcionales, que responden a ciertas ideas que tengo para aplicar este programa.

Como primer acercamiento quiero plantear este programa como una especie de juego de *agon* entre dos o más personas, para lo cual pienso hacer reuniones con músicos u otras personas interesadas en 'competir'. El desarrollo para lograr esto es bastante complejo, ya que requiere de módulos de: reproducción en tiempo real, herramientas gráficas y de informe más complejas, conectividad entre computadores y herramientas de automatización como por ejemplo la implementación de ondas para mantener un flujo constante en propiedades de los mensajes como el tempo (que tan rápido aparece el siguiente mensaje) o el timeout (que tanto dura en pantalla el mensaje). Una derivación de este acercamiento es utilizar el programa para crear unos *scripts* y ponerlos en acción de un computador contra otro.

Por otro lado también he venido desarrollando un modulo que conecta al programa con un sistema interactivo pantalla-cámara. La idea es proyectar los mensajes de error sobre los participantes de dicha interacción, acercando este proyecto al de los *Scripts para puntero*. Donde hay una negación al usuario, aunque también lo he pensado como un sistema donde los participantes pueden controlar a través del cuerpo 'el error'.

Por último, tengo pensados algunas muestras donde los mensajes de error se alimentan de diferentes fuentes (Internet, archivos en el computador, frases escritas, conjunción con otros programas generadores de sentido -¿por que no a Eliza o uno similar?) para que estos cumplan su función, tanto de afectar el sentido del espectador, como de negarlo e imponer el sentido de abolición.

Son varias las opciones y los acercamientos, y no puedo asegurar cual(es) se desarrollara(n) para la sustentación. También debo confesar que estas ideas son apenas unos indicios, unas semillas que creo pueden llegar a redondear la relación de sentido de los usuarios con aquello 'metafísico' en el computador. Sin embargo, esto no es algo que viene de antemano en la idea, sino que emerge en la medida que se va desarrollando. Es hasta posible que en este camino se presenten derivaciones y cambios.

# Bibliografía

---

## ***Libros, artículos y noticias***

- Caillois, Roger (1986). Los juegos y los hombres: la máscara y el vértigo. Traductor: Jorge Ferreiro. México: Fondo de Cultura Económica.
- Clarke, Arthur C. The Ultimate Machine, Harper's, Aug. 1958. En: Horten, Scott. Clark's Ultimate Machine. Harper's Magazine (Mar. 19, 2008). [11/05/2009]. Disponible en Web: <<http://www.harpers.org/archive/2008/03/hbc-90002682>>
- Dennett, Daniel (2004). La evolución de la libertad. Traductor: Ramón Vila Vernis. España: Paidós.
- Friedman, Ken; Smith, Owen y Sawchyn, Lauren (2002). the Fluxus Performance Workbook [en línea]. Performance Research e-Publications. Disponible en Web: <<http://www.thing.net/~grist/ld/fluxusworkbook.pdf> >
- Guatarri, Félix (1996). Caosmosis. Traductor: Irene Agoff. Buenos Aires: Ediciones Manatíal.
- Helluy, S. y Thomas, F. Effects of *Microphallus papillorobustus* (Platyhelminthes: Trematoda) on serotonergic immunoreactivity and neuronal architecture in the brain of *Gammarus insensibilis* (Crustacea: Amphipoda) [en línea]. The Royal Society. Proceedings: Biological Sciences, Vol. 270, No. 1515 (Mar. 22, 2003), pág. 563-568. [28/04/2009]. Disponible en Web: <<http://www.jstor.org/stable/3558702> >
- Man vs machine shocker: Kramnik allows mate in one [en línea]. Chessbase News (Nov. 27, 2006). [10/05/2009]. Disponible en Web: <<http://www.chessbase.com/newsdetail.asp?newsid=3509> >

- Ryle, Gilbert (2005). El concepto de lo mental. Traducción: Eduardo Rabossi. España: Paidós.
- Weizenbaum, Joseph. ELIZA--A Computer Program For the Study of Natural Language Communication Between Man and Machine [en línea]. [Mass., USA] MIT, Communications of the ACM Volume 9, Number 1 (January 1966), pág. 36
- 35. [25/10/2007]. Disponible en Web:  
<<http://i5.nyu.edu/~mm64/x52.9265/january1966.html> >
- Wittgenstein, Ludwig (1999). Investigaciones filosóficas. Traductores: Alfonso García y Ulises Moulines. España: Ediciones Altaya S. A.

### ***Películas***

- The Terminator (1984). Dirigida por James Cameron. Escrita por James Cameron y Gale Anne Hurd.
- Terminator 2: Judgement Day (1991). Dirigida por James Cameron. Escrita por James Cameron y William Wisher Jr.
- The Exorcist (1973). Dirigida por William Friedkin. Escrita por William Peter Blatty.

### ***Obras de arte***

- Meireles, Cildo. Inserciones y cultos ideológicos. [Intervención]. [1970-1976].

### ***Programas***

- Chess Base. Deep Fritz 11 - Multiprocessor Version. [4/05/2009]. Disponible en Web: < <http://www.chessbase.com/shop/product.asp?pid=> >
- Cross, Don. Chenard, a freeware chess program. [4/05/2009]. Disponible en Web: <<http://cosinekitty.com/chenard/> >
- Hayden, Charles. Eliza Test. [25/10/2007]. Disponible en Web: <<http://chayden.net/eliza/Eliza.shtml>>

- Johnson, Angus. Resource Hacker. Version 3.4.0. [26/04/2009]. Disponible en Web: < <http://www.angusj.com/resourcehacker/> >
- Mass, Christian. Freeware Hex Editor XVI32. Version 2.51. [28/04/2009]. Disponible en Web: <<http://www.chmaas.handshake.de/delphi/freeware/xvi32/xvi32.htm> >