

La sintaxis interactiva. Aportes de la semiótica a una teoría de la interacción social*

The interactive syntax Contributions of semiotics to a theory of social interaction

A sintaxe interativa Contribuições da semiótica para uma teoria da interação social

Carlos Scolari

Universitat Pompeu Fabra (Barcelona)

carlos.scolari@gmail.com

Fecha de recepción: 10 de enero de 2018

Fecha de recepción evaluador: 20 de enero de 2018

Fecha de recepción corrección: 30 de enero de 2018

Resumen

La semiótica juega un papel de suma importancia en la interacción social y en la interacción del usuario con la tecnología. Un campo de estudio que resultaría atractivo para la semiótica es la investigación a fondo de las instrucciones virtuales que se encuentran en los objetos que nos rodean, de este modo la interacción del usuario con la tecnología y las operaciones que realiza se volverían más rápidas, más fáciles, más eficientes, más atractivos y con un menor rango de error, logrando así la formación de una competencia interactiva entre usuarios.

Palabras clave: Sintaxis, interface, semiótica, operaciones, tecnología, usuarios.

Abstract

Semiotics plays a very important role in social interaction and user interaction with technology. A field of study that would be attractive for semiotics is the in-depth

investigation of the virtual instructions that are found in the objects that surround us, in this way the user's interaction with the technology and the operations he performs would become faster, more easy, more efficient, more attractive and with a lower range of error, thus achieving the formation of an interactive competition between users.

Keywords: Syntax, interface, semiotics, operations, technology, users.

Resumo

A semiótica desempenha um papel muito importante na interação social e interação do usuário com a tecnologia. Um campo de estudo que seria atraente para a semiótica é a investigação em profundidade das instruções virtuais que são encontradas nos objetos que nos cercam, desta forma a interação do usuário com a tecnologia e as operações que ele realiza se tornaria mais rápida, mais fácil, mais eficiente, mais atraente e com menor margem de erro, conseguindo assim a formação de uma competição interativa entre os usuários.

Palavras-chave: sintaxe, interface, semiótica, operações, tecnologia, usuários.

Las interfaces no están quietas. A veces se recombinan entre ellas, intercambiándose dispositivos y modalidades de interacción. Es así que un comando (o una secuencia de ellos) desarrollado originalmente para un videojuego puede aparecer en la enciclopedia multimedial o en un procesador de textos. Como los mitos de Lévy-Strauss, las interfaces "se piensan entre sí" y constituyen un sistema complejo de reenvíos y contaminaciones recíprocas.

Las interfaces, al recombinarse, se transforman y evolucionan. Si comparamos la lenta evolución de las interfaces de la escritura - desde el papiro al libro impreso, pasando por el *codex* medieval - con el desenfrenado desarrollo de las formas de interacción con los ordenadores, descubriremos que en un par de décadas se han quemado etapas a un ritmo vertiginoso. En menos de veinte años hemos pasado de la interfaz alfanumérica a las interfaces gráficas *Mac-style*, las cuales se están convirtiendo en ambientes *immersivos* proyectados para interacciones *totales*.

Si en las viejas interfaces alfanuméricas los comandos que el operador -un profesional altamente especializado que "hablaba" el lenguaje del sistema- debía digitar llevaron a una reflexión teórica sobre la *sintaxis de los lenguajes de programación*, con la difusión de las interfaces gráficas basadas en la *manipulación directa de objetos* a partir de 1984 -año de presentación del primer Macintosh- la situación cambió radicalmente:

por supuesto, existen aspectos sintácticos en la manipulación directa ... (pero) la sintaxis es limitada y puede ser pensada como perteneciente a los niveles más bajos de

las acciones dentro de la interfaz. Inclusive formas sintácticas como el doble-click o tener apretado el ratón y esperar parecen simples si las comparamos con las gramáticas de los primeros lenguajes de programación (Shneiderman, 1998, p. 61).

Sin embargo, que la sintaxis sea mucho más simple no significa que esté privada de aspectos interesantes para el análisis semiótico. A veces, detrás de dispositivos de interacción "ingenuos" o de interacciones "simples" se esconden complejos procesos perceptivos e interpretativos que permiten ir trazando el rumbo de lo que sería una *teoría semio-cognitiva de las interacciones con las máquinas digitales*.

Objetos y acciones, sintaxis e interacciones

Entendemos por *sintaxis de la interacción* una *secuencia de acciones* que el usuario de la interfaz debe ejecutar para lograr un objetivo determinado. La secuencia puede ser *única* -cuando el usuario está obligado a seguir un camino excluyente para llegar al resultado deseado- o, por el contrario, puede presentar diferentes alternativas. En este segundo caso la misma operación -por ejemplo modificar el estilo de un párrafo de texto o cambiar el contraste de una imagen- puede ejecutarse manipulando diferentes dispositivos hardware (teclado, ratón) o software (menú, íconos). Ya sea que se trate de crear y modificar una materia virtual -texto, imagen, sonido, etc.- hasta alcanzar el resultado deseado (software), de desplazarse dentro de un ambiente sintético para vencer un enemigo (videojuego) o de buscar una información (Internet), cada una de estas actividades exige la ejecución de *operaciones* que, en la mayor parte de los casos, recombinan y ordenan en *secuencias específicas* una *gama limitada de acciones*.

Desde un punto de vista sintáctico, los informáticos han identificado dos modelos de interacción entre el hombre y el ordenador. Estas dos formas de interacción, nacidas en los viejos ambientes alfanuméricos, todavía sobreviven en las más avanzadas interfaces gráficas:

Action-Object (*verbo-nombre*): consiste en elegir primero la acción (por ejemplo activando el comando "Abre") y después el objeto sobre el cual la acción se cumplirá (el documento a abrir, por ejemplo "interfaz.doc"). Este modelo no es muy flexible pero es el más indicado para los usuarios sin experiencia o que hacen un uso ocasional del sistema.

Object-Action (*nombre-verbo*): primero se elige el objeto y después se realiza la acción sobre él; en un programa de gráfica, por ejemplo, podemos seleccionar un rectángulo y llenarlo con un color o girarlo 90 grados. Este modelo de interacción es más flexible, deja libertad al usuario y precisamente por ese motivo es el más indicado para los operadores expertos.

Si las interfaces alfanuméricas de los primeros ordenadores de la posguerra -o en el más reciente sistema **DOS**- privilegiaban el modelo *action-object*, en las interfaces

gráficas difundidas a mediados de los años '80 el modelo *object-action* se transformó en la estructura de base para la proyectación de la interacción entre el hombre y los ordenadores. Ambos modelos, sin embargo, pueden convivir sin mayores contradicciones dentro de una misma interfaz. Sin ir muy lejos, en un programa como **WORD** encontramos ambas formas de interacción: para modificar un texto debemos primero seleccionarlo y después transformarlo (*object-action*); para importar una imagen debemos primero indicar la acción y después seleccionar la imagen (*action-object*). Algunos comandos que afectan a todo el documento -por ejemplo "Guarda"- eliminan la fase de selección del *object* y se limitan a la *action*.

Condensación/prolongación de la sintaxis

Los dos modelos de interacción presentados (*object-action* y *action-object*) pueden contener un número variable de operaciones. Por ejemplo en los primeros *word-processors* con interfaz alfanumérica la selección de una porción de texto (en otras palabras, la definición del *object*) exigía una larga secuencia de operaciones. Supongamos que un usuario de **WORD 3.2** (1986) debía mover cinco líneas más abajo un párrafo de texto compuesto por 10 líneas. Esta tarea -que se puede descomponer en la secuencia *seleccionar-cortar-mover-pegar-* exigía **26 acciones** a realizar con el teclado. La misma operación, ejecutada con el ratón, se resuelve en **4 acciones** (dos movimientos del ratón y activación de dos comandos -"Del" e "Ins"- presentes en la pantalla).

A partir de la versión **WORD 5.0** para **MACINTOSH** (1992) y de las sucesivas versiones para **WINDOWS** se le ofreció al usuario la posibilidad de seleccionar el texto con el cursor y llevarlo directamente hasta la nueva posición utilizando la técnica del *drag'n'drop*. Toda la secuencia se ha reducido a dos acciones prácticamente idénticas, muy "intuitivas" y en perfecta sintonía con la filosofía proyectual conocida como manipulación directa de objetos (en este caso textuales) (Shneiderman, 1998).

El *drag'n'drop*² constituye un claro ejemplo de *condensación de la secuencia de acciones*. Hay otros. Por ejemplo los proyectistas de los videogames de la saga de **MYST** (**MYST**, **RIVEN** y **EXILE**) han ideado un sistema para acelerar los desplazamientos del jugador *dentro de los ambientes ya visitados*. Cuando se pasa por segunda vez en un mismo lugar el jugador puede activar una función que permite un desplazamiento elíptico; en este caso la mano del cursor se transforma en una especie de rayo que señala la posibilidad de "saltar" desde el punto donde se encuentra hacia otro punto distante (Colombo-Eugeni, 1996). El ambiente que primero se había *explorado* con numerosas intervenciones del ratón, ahora se puede *atravesar* con un único "click". El mismo tipo de condensación lo encontramos en algunas webs dedicadas al *e-commerce*: por ejemplo un cliente ya registrado en **AMAZON.COM** puede acelerar la compra de un artículo con el sistema "1-Click" sin tener que volver a pasar por el proceso de inserción de datos personales, número de tarjeta de crédito, etc..

Pero si a los proyectistas de software o de webs comerciales la inclusión de formas condensadas de interacción les representa un valor agregado en sus productos o servicios³, en ciertas ocasiones los diseñadores *extienden de manera consciente las secuencias de interacción*. Volvamos al ejemplo del videojuego. Los proyectistas de **MYST** han *extendido las secuencias* para construir un *efecto de tensión* que actúa como "condimento" narrativo: "al aproximarse a una señal importante o a una revelación determinante, el avance rallenta estimulando y modulando la curiosidad del usuario" (Colombo-Eugeni,1996, p. 200). Un acercamiento que en otra parte del juego se resolvería con un único "click", en estos casos se descompone en varios acercamientos sucesivos que *frenan* el avance el jugador.

El proyectista de videojuegos, como un director de cine, juega permanentemente con la dimensión interactiva de su producto, acelerando o frenando -dentro de las posibilidades que brinda un *medium* que pasa gran parte del control al usuario- el tiempo de la narración a través de la manipulación de las secuencias de interacción.

La prolongación de la sintaxis y prevención de errores

El funcionamiento de la papelera ("Trash") sirve para delinear una serie de problemas que surge durante el uso de los dispositivos de interacción. Como hemos visto, para realizar cualquier operación el usuario debe respetar una *secuencia de acciones* predeterminada por el proyectista. Por ejemplo para cancelar definitivamente un documento el usuario debe seleccionarlo, "arrojarlo" dentro de la papelera y activar el comando "Vaciar papelera".

Los problemas surgen cuando el usuario, a pesar de haber respetado escrupulosamente una secuencia operativa proyectada por el diseñador, se equivoca y realiza una acción indeseada (por ejemplo, cuando cancela por error el contenido de la papelera). En esos casos el usuario debería poder retornar a la situación "pre-error". Según Don Norman:

si un error es posible, alguien antes o después lo cometerá. El proyectista debe partir del presupuesto de que todos los errores posibles serán cometidos, y preparar el proyecto en modo tal de reducir al mínimo las probabilidades de error o sus efectos una vez que se verificaron. Los errores deben ser fáciles de individualizar, deben tener consecuencias mínimas y, si es posible, sus efectos deben ser reversibles (1990, p. 52).

Uno de los dispositivos más cómodos y preferidos por los usuarios es seguramente el comando "Deshacer" ("Undo"), un mecanismo que permite anular la última operación realizada. Ulteriores desarrollos de este comando han llevado a la creación de paletas "cronológicas" -por ejemplo en **PHOTOSHOP**- que le permiten al usuario visualizar la secuencia de las operaciones realizadas y volver a algún momento anterior de la misma (*backtracking*).

Sin embargo en ciertas circunstancias -por ejemplo, cuando vaciamos la papelera- el comando "Undo" no funciona. Para prevenir que el usuario cometa errores irreparables los proyectistas han extendido algunas secuencias particularmente críticas incluyendo operaciones que no pueden ser "saltadas"; por ejemplo, antes de proceder a la cancelación definitiva de un documento que se encuentra dentro de la papelera, el ordenador exige al usuario la confirmación de la operación "Eliminar el documento ...". La secuencia de operaciones es, desde el punto de vista del proyectista, inobjetable. Para eliminar un *file* el usuario debe:

- Trasladar el documento dentro de la papelera
- Activar el comando "Vaciar la papelera"
- El sistema pregunta: "Confirma que desea eliminar ...?"
- Responder "SI" (si se desea cancelar el documento) o
- Anular" (si se desea cancelar la operación)
- "SI" - El documento es eliminado

Pero como todos sabemos, la creciente automaticidad de ciertas acciones durante la sesión de trabajo lleva a que este *dispositivo de seguridad* termine siendo "salteado" en manera inconsciente para ganar tiempo. Muchas veces el usuario confirma la eliminación del documento para arrepentirse unos pocos segundos más tarde, cuando la operación ya no tiene retorno ... Si bien estos dispositivos de seguridad basados en la *prolongación de las secuencias operativas* son necesarios en una primera fase de acercamiento a la interfaz, a medida que el usuario adquiere una mayor experiencia (y velocidad en la interacción) tienden a resultar inútiles porque *también su ejecución se vuelve automática*. Teniendo en cuenta esta tendencia a la automatización -que supera todas las vallas y trampas que el proyectista interpone para "dificultar" la interacción- la mayoría de los investigadores coincide en que *la interfaz ideal debería eliminar todas las acciones irreversibles*.

Breakdowns

Muchos *breakdowns* -entendidos como momentos de crisis que desembocan en una "caída" del sistema (Winograd-Flores, 1986)- se producen por un *incorrecto alineamiento de las secuencias operativas*. Veamos algunos ejemplos, dos provenientes del mundo "real" y el otro de una interfaz digital. En el aeropuerto de Caselle (Turín) existe un cajero automático de un banco que no mencionaremos (se dice el pecado pero no el pecador) que presenta la siguiente secuencia:

- Introducir la tarjeta magnética

- Introducir el código secreto
- Elegir (o digitar) la cantidad de dinero que se desea obtener
- Retirar el dinero
- Retirar la tarjeta magnética.

Esta secuencia operativa genera un error muy frecuente: el usuario -que entró en la cabina con el objetivo de *retirar dinero*- abandona el cajero una vez que obtuvo los billetes, dejando olvidada la tarjeta dentro de la máquina. Para solucionar un error de este tipo basta agregar otro paso (para poder salir del cajero es necesario abrir la puerta con la tarjeta) o, lo que es mucho mejor, modificar la secuencia (*primero* se devuelve la tarjeta y sólo *después* se entrega el dinero).

Volemos a otra ciudad. En los cajeros automáticos del Aeropuerto del Prat (Barcelona) la secuencia de acciones para pagar el estacionamiento es la siguiente:

- Introducir el ticket
- Introducir los billetes, monedas o tarjeta de crédito
- Retirar el ticket
- Retirar el cambio y el justificante de pago

Al igual que en el caso anterior, una vez que el usuario recibió el ticket validado por el cajero tenderá a considerar concluída la interacción (su objetivo era precisamente validar el ticket para poder salir del estacionamiento), olvidándose el eventual cambio y el justificante de pago. Un simple cambio en la secuencia de acciones (devolución del ticket validado sólo al final del proceso) bastaría para evitar inconvenientes.

Si en un *word-processor* (en ambiente Windows) estamos trabajando con un documento situado en el *floppy disk* (y no en el *hard disk*), es muy común que el disco sea extraído mientras el *file* sigue presente en la pantalla. Apenas el usuario intenta guardar el documento o cerrar la aplicación, la máquina "descubre" que el *floppy* no está más ... y exige la reintroducción del disco para poder completar la operación. En este caso sería necesario proyectar un mecanismo -como el que está presente en el sistema Macintosh⁴ - que evite la extracción del *floppy* mientras el documento sigue abierto en la pantalla, obligando al usuario a respetar la secuencia:

- Guardar documento
- Extraer floppy

Si bien no nos interesa desarrollar una *semiótica normativa* de las interacciones hombre-ordenador⁵, no podemos dejar de deducir algunos principios útiles para la proyectación interactiva. Es evidente que para construir una buena interfaz no basta descubrir una metáfora seductora o encontrar una buena solución gráfica: la *sintaxis de la interacción* -el orden lineal de las *secuencias operativas*- constituye uno de los aspectos fundamentales de este tipo de proyectación. Una sintaxis equivocada o contradictoria puede demoler la gráfica más fascinante o una consistente metáfora de interacción. La realización de una interfaz -y esto no es una novedad- exige competencias que van mucho más allá del diseño gráfico o la programación, ya que se debe incorporar en la proyectación una dimensión hasta ahora prácticamente ausente en el *graphic design* tradicional: *el tiempo*⁶.

Un paquete de instrucciones pragmáticas

Como nos enseña el *modelo instruccional* de la semiótica interpretativa (Eco, 1979 - Bettetini, 1984) un texto no sólo se presenta como un "paquete" de instrucciones semánticas: también propone a su lector/espectador "un intercambio comunicativo simbólico, una 'conversación' simbólica. La representación de sus valores semánticos está a su vez enmarcada en la representación de su translación entre dos sujetos que el aparato textual constituye, como imágenes fantasmagóricas, de quien transmite y de quien recibe". En otras palabras, el texto también "representa sus normas de uso, sus modalidades de acceso al sentido" (Bettetini, 1984, p. 101) a través de un paquete de instrucciones pragmático-comunicacionales. La aplicación del modelo instruccional a las interfaces digitales abre un campo de investigación hasta ahora poco frecuentado. Las mejores interfaces, las más fáciles de usar, no son necesariamente aquellas que nos impactan gráficamente sino las que vehiculizan un paquete de instrucciones pragmáticas coherente y simple.

Guionando la interacción

El diálogo entre la semiótica y las ciencias cognitivas ha promovido un interesante intercambio de saberes. Una obra como "LECTOR IN FABULA" (Eco, 1979) sería impensable sin el soporte conceptual de los teóricos de los procesos de conocimiento. Por ejemplo los marcos (*frames*) de Marvin Minsky y los guiones (*scripts*) de Roger Schank le sirvieron a Eco como "barras de grafito" -la metáfora es de Paolo Fabbri- para evitar la explosión de la "semiosis ilimitada" y frenar la "fuga de los interpretantes"⁷. Los textos, sostiene Eco, no son interpretables infinitamente, a un cierto punto hay que frenarse; si se va más allá no estamos "interpretando" sino "usando" un texto (1979; 1984; 1991)⁸. También Teun van Dijk ha recuperado el concepto de *frame* -entendido como "formas de organización del conocimiento convencionalmente establecido que poseemos del mundo (1996:185)- dentro de su semiótica de impronta pragmática.

Un "frame" puede ser descrito como una estructura de informaciones que sirve para representar una situación estereotipada. Según Minsky:

cada experiencia perceptiva activa ciertas estructuras llamadas frames, estructuras que hemos adquirido en el curso de la experiencia pasada. Todos recordamos millones de estos frames, cada uno de los cuales representa una situación estereotipada, como encontrar un cierto tipo de persona, encontrarse en un cierto tipo de habitación o participar en un cierto tipo de recepción (1989, p. 477).

Los guiones (*scripts*), sostiene por su parte Roger Schank, son "secuencias de cadenas casuales preempaquetadas" (1987, p. 140). Estas estructuras mentales influyen sobre todos nuestros pensamientos y acciones. Incluso de frente a situaciones desconocidas no podemos hacer a menos de recurrir a estas estructuras; en este caso la mente recupera algunos de los *frames* y *scripts* archivados en la memoria, los adapta y recombina a la nueva experiencia. La mente elige determinadas "escenas" y las "monta" linealmente para construir un *programa de acción*⁹.

Programas de acción

Volviendo a la aplicación semiótica de los conceptos de *frame* y *script*, Eco describe a los guiones (*sceneggiature*) como:

esquemas de acción y de comportamiento preestablecidos (como el participar en una fiesta, ir a la estación a tomar un tren, servir y consumir una hamburguesa). Además de los guiones comunes se pueden concebir guiones intertextuales, o reglas de género (por ejemplo, como debería verosimilmente concluir el duelo entre el sheriff y el bandido en un western tradicional) (1984, p. 71).

La *enciclopedia* del lector registra bajo forma de guión no sólo las instrucciones para la inserción de los signos en los diferentes contextos, sino también las reglas pragmáticas que rigen la acción interpretativa de los sujetos. De la misma manera, las interacciones con los objetos que nos rodean han sido formalizadas por nuestra memoria y archivadas bajo forma de guión: abrir una ventana o destapar una botella son todos *programas de acción* registrados en nuestra enciclopedia. Una operación muy simple como la apertura de la puerta de un coche puede ser subdividida en una secuencia de micro-acciones:

- Colocar la llave
- Girar la llave
- Retirar la llave
- Apretar la manija
- Tirar de la manija hacia afuera

Las interacciones con el ordenador siguen el mismo recorrido de las interacciones con los objetos del mundo "real": para efectuar un "cut'n'paste" ("corta y pega") el usuario de la interfaz debe ejecutar una serie precisa de micro-acciones:

- Seleccionar el elemento
- Ejecutar el comando "Corta" (el elemento seleccionado desaparece)
- Colocar el cursor en el lugar adonde se quiere pegar el elemento
- Ejecutar el comando "Pega" (el elemento aparece en la nueva posición)
- Si el usuario de un programa de gráfica desea colorear una figura, debe:
 - Seleccionar la figura
 - Abrir la paleta de colores
 - Seleccionar el color
 - Clicar "OK"

Todas estas secuencias operativas -caracterizadas en estos casos por la sintaxis *object-action*- quedan registradas en la memoria del usuario bajo forma de guión. Un *script* de este tipo (que puede ser reducido a la secuencia "Seleccionar - Modificar - OK") puede considerarse una *programa de acción estándar*, una especie de *principio organizador primario y transversal que incluye todas sus posibles variantes* (la misma secuencia también puede ser aplicada si deseamos cambiar la dimensión de un carácter en **WORD** o el contraste de una fotografía en **PHOTOSHOP**). Según el psicólogo Mantovani:

una vez aprendida una secuencia de comandos, se tiende a usar sólo esa, inclusive cuando existen a disposición procedimientos más simples o potentes. Estos no son explorados porque (el usuario) se conforma con el resultado, mediocre pero aceptable, que se consigue con la rutina ya conocida (1995, p. 67).

La fuerza cognitiva de estos mini-programas de interacción -cuya actualización es realizada por el usuario en manera *casiautomática*- hace posible la aparente *transparencia de la interacción*.

Se busca guionista (inútil presentarse sin experiencia)

¿Qué sucede cuando el usuario de un ordenador se encuentra de frente a un dispositivo desconocido? Como el lector de una novela policial, que debe resolver un crimen aplicando toda su experiencia de lector de narrativa de género, el usuario recuperará de su enciclopedia los *scripts* que parecen adaptarse mejor a la situación. Los

guiones son estructuras flexibles, que permiten la recuperación de las experiencias previas de interacción para interpretar las situaciones novedosas. Desde esta perspectiva el trabajo del *interface designer* tiene poco de programación y mucho de semiótica: el proyectista debe rescatar experiencias concretas de interacción "guionizadas" por el usuario -como girar manijas, apretar botones, mover palancas, etc.- e incorporarlas en los ambientes interactivos.

Muchos conceptos utilizados por los proyectistas de interfaces adquieren una nueva dimensión bajo la luz de la perspectiva semio-cognitiva. Por ejemplo la "usability" -la facilidad de uso de un dispositivo interactivo (Nielsen, 1993, 2000; Rubin, 1994)- se vincula con la cantidad y la calidad de las *instrucciones pragmáticas* que la interfaz contiene bajo la forma de un programa de acción virtual y replegado. Más claros y económicos serán los guiones que la interfaz activa en la mente del usuario, *más transparente y automática parecerá la interacción*. Se podría decir que *todos los objetos -virtuales o reales que sean- poseen un programa de acción replegado en espera de un usuario*. El gran desafío del *interaction designer* consiste precisamente en proyectar el espacio de encuentro entre estos programas virtuales y el bagaje de experiencias del usuario archivado en su memoria bajo forma de *frames* y *scripts*.

Los guiones y la dinámica proyección/uso

Según Eco los guiones pueden ser considerados "textos virtuales condensados". La comprensión textual está "ampliamente dominada por la aplicación de guiones pertinentes, así como las hipótesis textuales destinadas al fracaso dependen de la aplicación de guiones equivocados e infelices" (1979, p. 81). Como apenas hemos visto, la interpretación de una interfaz está siempre vinculada a la aplicación de la "historia" más adecuada por parte del usuario. A menudo los *breakdowns* se producen porque *el usuario aplica un guión equivocado*.

Hace algunos años se instalaron en Francia nuevos parquímetros que funcionaban con una tarjeta magnética pre-pagada. La secuencia era por demás simple; el automovilista debía:

- Digitar el tiempo de permanencia estimado
- Introducir la tarjeta magnética
- Retirar la tarjeta magnética
- Retirar el ticket

Poco tiempo después el sistema se bloqueó; las máquinas fueron quedando progresivamente fuera de servicio porque los usuarios no respetaban la secuencia prevista por los proyectistas. Después de una breve investigación se descubrió por qué un

mecanismo tan simple había causado tantos inconvenientes: los usuarios aplicaban al parquímetro el guión -ya registrado en su enciclopedia- correspondiente a los teléfonos públicos, donde *primero* se coloca la tarjeta y *después* se digita el número. La secuencia de micro-acciones de frente al parquímetro se resolvía en un gesto *automático*: una operación que parecía *transparente* al usuario dejaba de serlo en cuanto el sistema no respondía como previsto¹⁰.

La distancia que existe entre lo que el proyectista de interfaces se espera del sistema que ha creado (y del usuario que lo utilizará) y los guiones que el usuario real aplica durante la interacción parece ser uno de los motores que dinamiza la evolución de las interfaces. Para superar este desequilibrio los proyectistas construyen nuevas versiones de sus productos, las que a su vez serán sometidas a usos *desviados* y totalmente inesperados debido a la aplicación de guiones errados por parte de los usuarios. El uso - escribe Patrice Flichy-

permite al operador mejorar la performance de la máquina; se perfila de este modo una mejor adecuación entre la máquina y el hombre. El uso puede igualmente llevar a modificaciones de la máquina. Según su relevancia, a hacerse cargo será el usuario o el productor (1996, p. 26).

Al desviarse de los usos previstos por el proyectista -hace veinte años se hubiese dicho: al "decodificar de manera aberrante la interfaz"- el usuario empírico está efectuando un trabajo creativo, una re-proycción de la interfaz que, antes o después, terminará repercutiendo sobre la misma evolución del producto digital.

Instrucciones de uso (Y ahora qué?)

Los proyectistas de interfaces suelen repetir una frase que viene a nuestro caso: "la mejor interfaz es aquella que no necesita manual de instrucciones". La mejor interfaz, agregamos nosotros, es la que tiene el manual "incorporado" en sí misma. Veremos a continuación algunas situaciones sintácticas clásicas (verdaderas *encrucijadas* para el usuario) donde las instrucciones juegan un papel fundamental. En todos estos casos las instrucciones tratan de responder a la siguiente pregunta del usuario: *qué puedo (debo) hacer a continuación?*

Alternativas binarias / Alternativas múltiples Veamos el caso más simple analizando un ejemplo ya presentado en la sección 1.2. Para eliminar definitivamente un documento dentro de la papelera el usuario debe:

- Activar el comando "Vaciar la papelera"
- El sistema pregunta: "Confirma que desea eliminar ...?"
- Responder "Si" (si se desea cancelar el documento) o

- Anular" (si se desea cancelar la operación)

En este caso el usuario está obligado a elegir entre dos opciones: continuar el camino trazado por la sintaxis prevista por el proyectista o salir de él, interrumpiendo el proceso para volver a una fase anterior del mismo. El usuario no puede realizar ninguna otra operación con el ordenador -en otras palabras, no podrá salir de ese estado- hasta que no concluya (o anule) la secuencia en acto. En términos modales estas *alternativas binarias* se colocan en el orden del *deber-hacer* (o una o la otra).

A las alternativas binarias se oponen, obviamente, las secuencias interactivas que ofrecen al usuario un amplio espectro de posibilidades (por ejemplo un software o un portal con infinidad de links a otras páginas y webs). Las denominaremos *alternativas múltiples*.

Alternativas independientes del estado / Alternativas dependientes del estado

Imaginemos que un navegante de la red digital llega al mapa de una web: tendrá a su disposición una gran cantidad de links, todos ellos disponibles para ser activados y "llevarlo" a la página seleccionada. Cualquier navegante que entre en el mapa, más allá de su proveniencia o recorrido precedente, verá los mismos links y tendrá las mismas posibilidades de navegación. Las *alternativas*, en este caso, serán *independientes del estado* (del sistema) y del recorrido precedente del usuario.

Imaginemos ahora el usuario de un software de *word-processing* que desea modificar una tabla. Ahora bien, *si antes no selecciona la tabla, los comandos del menú "Tabla" aparecerán en color gris y, por lo tanto, no podrán ser activados*. El código de interacción es explícito: las instrucciones presentes en la interfaz "le están diciendo" al usuario que esos comandos permanecerán inactivos hasta que no seleccione una tabla. De esta manera el software elimina algunas alternativas de interacción y habilita otras, limitando las posibilidades de desarrollo de una determinada secuencia. Dicho en términos lingüístico-informacionales: las posibilidades de *saturación* de esa secuencia son menores respecto al ejemplo anterior. Se trata en este caso de *alternativas dependientes del estado* (del sistema) y del recorrido precedente del usuario.

En términos modales las *alternativas independientes del estado* se colocan en el orden del *poder-hacer* mientras que las *alternativas dependientes del estado* juegan con la dialéctica del *poder-hacer / no poder-hacer*.

Instrucciones presentes / Instrucciones ausentes

Hasta ahora hemos visto instrucciones explícitas que pertenecen al orden del *hacer-saber*. Como ya vimos, la mejor interfaz es aquella que contiene en sí misma su manual de instrucciones, el cual se presenta bajo forma de *programa de interacción replegado* que el usuario actualiza durante el proceso de interacción. Sin embargo, existe

un tipo de texto digital que *esconde sus instrucciones*: los videojuegos. Para lograr un *efecto de extrañamiento*, los diseñadores suelen obligar al jugador a descubrir la *gramática de interacción* que caracteriza a cada *videogame*, ocultando su *código* y las posibles secuencias operativas. Se trata de productos proyectados para la *exploración*, en los que el usuario debe *reconstruir paso a pasouna gramática* siguiendo el método de la prueba/error. Más que la resolución de un enigma o la derrota de un enemigo, la pregunta con la que abrimos esta sección -*Qué puedo (debo) hacer a continuación?*- es quizás el principal combustible que alimenta el motor narrativo de los juegos digitales. En términos modales, los dispositivos de interacción que prevalecen a estos textos -caracterizados por las *instrucciones ausentes* o, cuanto menos, escondidas- se colocan en el orden del *no hacer- saber*¹¹.

La dimensión narrativa de la interacción

La importancia de la dimensión narrativa de los discursos parece ser un indiscutible punto de encuentro entre los diferentes filones que animan el debate semiótico. Muchos investigadores están convencidos de que la narratividad es el principio mismo de la organización de cualquier discurso. Según Greimas existen "formas narrativas casi universales", una especie de "competencia narrativa" de carácter virtual presupuesta a la producción y lectura de los discursos. Para el semiótico lituano "todo discurso es narrativo" (Greimas, 1985, p. 16). También la semiótica interpretativa ha repetidamente reconocido la universalidad de las formas narrativas. Según Eco es posible encontrar una estructura narrativa en cualquier texto, incluso en aquellos que aparentemente no lo son.

Si bien en cierto tipo de producción digital -por ejemplo en los *videogames* o en las "novelas interactivas" como "AFTERNOON" de Michael Joyce- la dimensión narrativa es más que evidente, lo "contado" está presente en cualquier experiencia de interacción con los ordenadores. Qué es la interacción sino un proceso en el cual un *sujeto*, a través de la manipulación de una serie de *instrumentos*, transforma un entorno generando una sucesión de *estados* (siguiendo un eje lógico-temporal) hasta obtener el *objetivo deseado*? En algunos casos la narración no implica necesariamente un acto de creación -por ejemplo cuando se crea una imagen con un programa de gráfica (Bettetini, 1991)- sino la búsqueda de una información dentro de una red hipermedial de datos. En ambos casos nos encontramos, como sostenía Greimas, con "sujetos que corren detrás de objetos"

El reconocimiento de la dimensión narrativa en el universo de la interactividad hombre-ordenador abre interesantes perspectivas para la investigación semiótica. Seguramente una mayor aplicación de las categorías narratológicas daría buenos resultados en el ámbito de la HCI. Entre otras cosas, nos permitiría ir mucho más allá del cruce entre semiótica y ciencia cognitiva que aquí hemos esbozado.

Como una buena novela, los ambientes interactivos bien proyectados poseen una enorme capacidad para "capturar" al usuario. Las interfaces y los procesos de interacción pueden contribuir a la creación de estados afectivos y, como cualquier otro texto, suscitan *pasiones*. Un análisis más profundo de la *dimensión pasional* de las interacciones con las máquinas digitales quizás nos permitiría entender por qué ciertos productos (webs o videogames) alcanzan gran éxito a pesar de presentar evidentes problemas de usabilidad¹².

Como vemos, el cruce entre semiótica y ciencia cognitiva permite enriquecer el análisis de procesos aparentemente simples que esconden no pocos vericuetos interpretativos. ¿Qué puede aportar la semiótica a un campo de investigación hasta ahora hegemonizado por la ciencia cognitiva y la psicología? Creemos que sólo una lectura en clave semiótica de los procesos de interacción nos permitirá superar el *instrumentalismo* presente en las actuales teorías de la HCI, que terminan siempre considerando a la interfaz una simple *extensión o prótesis* del cuerpo y a la interacción una actividad *natural*. Se trata, en definitiva, de desmontar los complejos procesos semióticos que se esconden detrás de la aparente automaticidad de la interacción para evidenciar sus aspectos *contractuales e interpretativos*.

Referencias

- Anceschi, G. (ed.) (1992) *Il Progetto Delle Interfacce*. Milán: Domus Academy.
- Bettetini, G. (1984) *La conversazione audiovisiva*. Milán: Bompiani.
- Bettetini, G. (1991) *La simulazione visiva*. Milán: Bompiani.
- Briggs, P. (1988) What We Know And What We Need To Know: The User Model Versus The User's Model In Hci. In *Behaviour And Information Technology*, vol. 7 - n° 4. London: Taylor and Francis Ltd, p. 431.
- Carroll, J. (ed.) (1987) *Interfacing Thought - Cognitive Aspects Of Human-Computer Interaction*. Cambridge: MIT Press.
- Colombo, F. & Eugeni, R. (1996) *Il testo visibile - Teoria, storia e modelli di analisi. La Nuova Italia Scientifica*. Milán.
- Van Dijk, T. (1978) *Tekstwetenschap. Een interdisciplinaire inleiding*. Holanda: Het Spectrum.
- Eco, U. (1979) *Lector in fabula*. Milán: Bompiani.
- Eco, U. (1984) *Semiotica e filosofia del linguaggio*. Milán: Bompiani.

- Eco, U. (1994) *Sei passeggiate nei boschi narrativi*. Milán: Bompiani.
- Eco, U. (1995) *Interpretazione e sovrainterpretazione*. Milán: Bompiani.
- Eco, U. (1997) *Kant e l'ornitorinco*. Milán: Bompiani.
- Flichy, P. (1995) *L'innovation technique*. París.
- Gibson, J. J. (1987) *The Ecological Approach To Visual Perception*. Lawrence Erlbaum Assoc.
- Greimas, A. (1983) *Du sens II*. París.
- Landowski, E. & Marrone, G. (eds.) (2002) *La società degli oggetti*. Roma: Meltemi.
- Mantovani, G. (1995) *L'interazione uomo-computer*, II. Mulino: Boloña.
- Minsky, M. (1985) *The Society Of Mind*.
- Nacci, M. (ed.) (1998) *Oggetti d'uso quotidiano*. Venecia: Marsilio.
- Nielsen, J. (1993) *Usability Inspection Methods*. Boston: AP Professional.
- Nielsen, J. (2000) *Designing Web Usability*. Indiana: New Riders Publishing.
- Norman, D. (1990) *The Design Of Everyday Things*. New York: Doubleday.
- Norman, D. (ed.) (1981) *Perspectives On Cognitive Sciences*.
- Raskin, J. (2000) *The Human Interface - New Directions For Designing Interactive Systems*. New York: ACM Press - Addison-Wesley.
- Rubin, J. (1994) *Handbook Of Usability Testing*. New York: John Willey & Sons.
- Schank, R. (1987) *Lenguaje y memoria*. Norman, D (ed.), p. 129.
- Schneiderman, B. (1997) *Designing The User Interface: Strategies For Effective Uman-Computer Interaction*. Massachusetts: Addison-Wesley.
- Scolari, C. (1995) *Hyperbook - Un Iper testo Per Sapere Cosa Sono Gli Iper testi*. Boloña: Synergon.
- Scolari, C. (2001) Per Un Pugno Di Iperlibri. Iperfiction, Narrativa E Retorica Iper testuale. En Bertetti & Mannetti (eds.) *Le Forme Della Testualità*. Turín: Testo & Immagine.
- Scolari, C. (2001) Towards A Semio-Cognitive Theory Of Hci. En *Extended Abstracts - Conference On Human Factors In Computing Systems*. New York: ACM Press.

Semprini, A. (ed.) (1999) *Il Senso Delle Cose*. Milán: Francoangeli.

Winograd, T. & Flores, F. (1986) *Understanding Computers And Cognition*.

Notas

* Publicado en el número 35 de *Razón y Palabra*. Disponible en: <http://www.razonypalabra.org.mx/anteriores/n35/cscolari.html>

2 El empleo de este dispositivo de interacción nacido en los programas de DTP (Desktop Publishing) dentro de un programa de word-processing confirma una vez más el fuerte proceso de contaminación e hibridación que anima el universo de las interfaces digitales.

3 Uno de los postulados máximos de esa ideología proyectual conocida como Web Usability es precisamente la velocidad (Nielsen, 2000).

4 El sistema Macintosh presenta una lógica totalmente diversa: para extraer un floppy disk o un cd-rom el usuario debe trasladarlo con el cursor sobre la papelera aplicando la técnica del drag'n'drop. Si un documento presente en el disco está abierto el floppy (o el CD-ROM) no pueden ser extraídos; si trasladamos el ícono del disco sobre la papelera un mensaje nos informa que el file está en uso y que por lo tanto no puede ser "expulsado" del ordenador.

5 La Human-Computer Interaction (HCI) es un campo de estudio en el cual la reflexión teórica y la investigación empírica siempre han estado obligadas a ofrecer consejos y "guías" destinadas a los proyectistas de sistemas informáticos (no es casual que la mayor parte de los artículos y libros dedicados a la HCI terminen con la clásica lista de consejos y "guidelines"). Quizás para los investigadores de la cognición este "valor agregado" se presente como un hecho natural y descontado. Respecto a la semiótica, no vemos la necesidad de implementarlo: el objetivo de la semiótica no es ofrecer principios o guidelines para mejorar las interfaces sino construir modelos teóricos para comprender mejor la interacción hombre-ordenador. Si en un segundo momento los modelos elaborados por la semiótica son retomados por los proyectistas y resultan útiles para la realización de mejores interfaces -o para optimizar los procesos de interacción- es otra cuestión que supera la pertinencia de la intervención semiótica.

6 En este contexto comienza a delinearse una nueva figura profesional -el interaction designer-, un proyectista que recoge la experiencia del diseño tradicional y la recombina con competencias propias de los dominios donde la variable temporal ocupa un lugar central (por ejemplo el trabajo del director teatral, musical o cinematográfico) (Anceschi, 1993).

7 Dentro de la semiótica interpretativa de Eco los marcos y los guiones constituyeron la principal barrera para contener no tanto la fuga de los interpretantes sino los ataques impetuosos de las tropas deconstruccionistas, las cuales enarbolaban las banderas de la deriva infinita del sentido allá por los años '80.

8 Sin renunciar a la pertinencia semiótica, en los últimos años Eco se ha acercado todavía más a la ciencia de la cognición; en una de sus más recientes recopilaciones de "exploraciones vagabundas" y de apuntes teóricos -"KANT E L'ORNITORINCO" (1997)- indaga precisamente en esa zona donde los procesos perceptivos comienzan a transformarse en interpretación. Sin embargo, Eco no pretende "meter las narices en la caja negra de nuestros procesos mentales o cerebrales. Me pondré sólo algunas preguntas

sobre las relaciones entre un posible neo-esquematismo y las nociones semióticas de significado, de diccionario e enciclopedia y de interpretación" (1997:105). En nuestro caso no podremos hacer a menos de realizar breves incursiones dentro de la "caja negra" para tratar de entender mejor los procesos semióticos y cognitivos que se manifiestan durante la interacción con las máquinas digitales.

9 En realidad los guiones no son estructuras presentes dentro de la memoria en manera integral sino un conjunto de diferentes subpartes que, de frente a una situación determinada, son "llamadas" y recombinadas para poder formar la "cadena casual empaquetada". Este proceso reconstructivo -que demuestra la existencia de una precisa economía de la memoria- funciona con la misma lógica del montador cinematográfico de frente a la moviola: la mente elige determinadas escenas y las coloca linealmente para construir un programa de acción.

10 Veamos otro error muy común proveniente de la prehistoria de los Personal Computers. En su primer contacto con un word-processor el usuario inexperto recuperaba de la memoria una serie de guiones provenientes de su experiencia de interacción con dispositivos muy parecidos, en primer lugar, las máquinas de escribir. Una serie de comandos e íconos tendían a reforzar la aplicación de estos guiones que le facilitaban notablemente el aprendizaje del software. Sin embargo, el guión de la máquina de escribir también limitaba el uso del word-processor; por ejemplo, el usuario inexperto tendía -al final de cada línea- a llevar el cursor al inicio de la siguiente digitando la tecla "Enter" como si se tratase del carro de la máquina (cuando en realidad el ordenador realizaba la operación automáticamente) (Briggs, 1988). El usuario, obviamente, no sabía que la operación era realizada por la computadora automáticamente. Como podemos observar, hasta un cierto punto el modelo de la máquina de escribir "funciona", llegando a facilitar notablemente el aprendizaje del software para el tratamiento textual. Pero la aplicación de un modelo hasta sus últimas consecuencias (¡el word-processor simula una máquina de escribir, pero no es una máquina de escribir!) puede entorpecer el proceso de interacción llevando a la aplicación de guiones errados.

11 Esta breve presentación de algunas situaciones permite vislumbrar para la semiótica un interesante campo de estudio: la investigación de las instrucciones virtuales presentes en los objetos que nos rodean. Un trabajo de este tipo -hasta ahora parcialmente realizado con algunos objetos del "mundo real" (Nacci, 1998; Semprini, 1999; Landowski-Marrone, 2002)- debería partir de una clasificación de las diferentes instrucciones presentes en los ambientes digitales, para ir avanzando de a poco hacia las complejas articulaciones que regulan la formación de una competencia interactiva en los usuarios.

12 Una semiótica de las interacciones podría transformarse en un (otro) punto de encuentro entre la semiótica interpretativa, marcada por la impronta cognitiva y la semiótica generativa, más atenta al espacio patémico.