

<http://artnodes.uoc.edu>

## ARTÍCULO

### NODO «ARTE, CULTURA Y CIENCIAS DE LA COMPLEJIDAD»

# Life Spacies y Life Spacies II: modelar sistemas complejos para arte interactivo

**Christa Sommerer y Laurent Mignonneau**

Fecha de presentación: octubre de 2009

Fecha de aceptación: octubre de 2009

Fecha de publicación: diciembre de 2009

## Resumen

Basándonos en la idea de que la interacción y la comunicación entre entidades de un sistema son las fuerzas que impulsan la aparición de estructuras más complejas y elevadas en la vida, proponemos aplicar principios de la teoría de sistemas complejos a la creación de obras de arte interactivas, generadas por ordenador y basadas en la participación del público, y comprobar si la complejidad puede surgir en un sistema artificial generado por ordenador.

## Palabras clave

Life Spacies, arte interactivo, modelado, teoría de sistemas complejos

## Abstract

*Based on the idea that interaction and communication between entities of a system are the driving forces behind the emergence of higher and more complex structures in life, we propose applying the principles of complex system theory to the creation of interactive, computer-generated and audience-participatory artwork and testing whether complexity within an artificial computer-generated system can emerge.*

## Keywords

*Life Spacies, interactive art, modelling, complex system theory*

## 1. Introducción

La creación de vida virtual en ordenadores acaba planteando la cuestión de cómo surgió la vida en la tierra y de cómo podría haberse desarrollado partiendo de unidades o partículas más simples hacia estructuras cada vez más complejas o sistemas completos de estructuras que parecen seguir una cierta pauta interna de organización. Ésta es también la pregunta clave en la nueva «ciencia de sistemas complejos». La primera parte de este artículo analiza algunas de las teorías y principios de sistemas complejos y luego propone aplicarlos a la creación de una obra de arte interactiva, generada por ordenador y basada en la participación del público.

## 2. Ciencia de sistemas complejos

La ciencia de sistemas complejos ha surgido como campo de investigación en la última década. Estudia «cómo las partes de un sistema suscitan los comportamientos colectivos de ese sistema y cómo interactúa el sistema con su entorno. Los sistemas sociales formados (parcialmente) por personas, el cerebro formado por neuronas, las moléculas formadas por átomos y el tiempo meteorológico que se forma a partir de corrientes de aire constituyen ejemplos de sistemas complejos. El campo de los sistemas complejos atraviesa todas las disciplinas tradicionales de la ciencia así como la ingeniería, la gestión empresarial y la medicina. Se centra en ciertas preguntas sobre partes, todos y relaciones. Estas preguntas son relevantes para todos los campos tradicionales [...] Hay tres enfoques interrelacionados para el estudio moderno de los sistemas complejos: 1) cómo las interacciones suscitan pautas de comportamiento, 2) de qué maneras se pueden describir los sistemas complejos y 3) el proceso de formación de sistemas complejos a través de la formación y evolución de pautas.»<sup>1</sup>

## 3. Definir la complejidad

Aunque no hay una definición exacta de lo que es un sistema complejo, hoy en día se asume que cuando un conjunto de partículas o agentes autónomos en evolución interactúan, el sistema global resultante muestra propiedades colectivas emergentes, una evolución y un comportamiento crítico que poseen características universales.

Estos agentes o partículas pueden ser moléculas complejas, células, organismos vivos, grupos de animales, sociedades humanas, empresas industriales, tecnologías que compiten, etc. Todas ellas son agregaciones de materia, energía e información que muestran las siguientes características:

- Se vinculan entre sí.
- Aprenden, se adaptan y organizan.
- Mutan y evolucionan.
- Expanden su diversidad.
- Reaccionan ante sus vecinos y el control externo.
- Exploran sus opciones.
- Se reproducen.
- Se organizan en una jerarquía de estructuras de orden superior.

Hallar un principio común tras las fuerzas organizativas de los sistemas naturales resulta una tarea compleja, y parece que haya tantas teorías como teóricos. Mencionaremos brevemente algunas de las numerosas teorías sobre sistemas complejos. Edmonds<sup>2</sup> nos ha permitido extraer información valiosa sobre los diversos enfoques y definiciones de teoría de sistemas complejos.

### 3.1. Complejidad de información algorítmica (AIC) – La definición de KCS

La definición más conocida de complejidad es la de KCS (Kolmogorov-Chaitin-Solomonoff),<sup>3</sup> que describe la complejidad de información algorítmica (AIC) situándola en algún punto entre el orden y la aleatoriedad; es decir, que la complejidad aumenta cuando  $P_{min}$  (el algoritmo más corto que puede generar una secuencia de dígitos,  $S$ ) alcanza una extensión equivalente a la de la secuencia que hay que procesar. Cuando el algoritmo alcanza este límite de incompreibilidad, la secuencia se define como aleatoria. La definición de KCS distingue entre estructuras «altamente ordenadas» y «altamente complejas».

### 3.2. Definición del número de partes de Hinegardner y Engelberg

Puede que la medida más sencilla de complejidad sea la sugerida por Hinegardner y Engelberg:<sup>4</sup> el número de partes distintas. La medida

- 
1. BAR-YAM, Yaneer (2000). «What is the study of complex systems?» [artículo en línea]. *New England Complex System Institute*. <<http://necsi.org/guide/whatis.html>>
  2. EDMONDS, B. (1999). *Syntactic Measures of Complexity*. Tesis doctoral presentada en la Universidad de Manchester, Reino Unido.
  3. KOLMOGOROV, A.N. (1965). «Three Approaches to the quantitative definition of Information». *Problems of Information Transmission*. N.º 1, págs. 1-17; SOLOMONOFF, R.J. (1964). «A Formal Theory of Inductive Inference». *Information and Control*. N.º 7, págs. 1-22, 224-254; CHAITIN, G. J. (1966). «On the Length of Programs for Computing Finite Binary Sequences». *Journal of the Association of Computing Machinery*. N.º 13, págs. 547-569.
  4. HIINEGARDNER, R.; ENGELBERG, H. (1983). «Biological Complexity». *J. Theoret. Biol.* N.º 104, págs. 7-20.

de Hinegardner y Engelberg nos recuerda los diagramas «expandidos» de las piezas de maquinaria. Ofrecen algunas indicaciones de complejidad pero obvian lo que quizás sea lo más importante: la «organización» y los «niveles de organización».<sup>5</sup>

### 3.3. Complejidad topológica de J. Crutchfield

La complejidad topológica descrita por Crutchfield<sup>6</sup> es una medida del tamaño del modelo numérico mínimo (que suele ser alguna clase de autómatas finito) en el lenguaje formal mínimo en el que posee un modelo finito. Así que la complejidad del modelo se «objetiva» al considerar sólo modelos mínimos pero también se vincula a la jerarquía fija de los lenguajes formales.

### 3.4. Complejidad computacional

Hoy en día la complejidad computacional es un área muy estudiada con muchos resultados formales.<sup>7</sup> La teoría de la complejidad se basa en la investigación sobre teoría de la computabilidad llevada a cabo a partir de los años treinta por, entre otros, Alan Turing, Alonzo Church y Stephen Kleene. En aquel entonces, las consideraciones primordiales eran la formalización de la noción de ordenador (como la máquina de Turing o el cálculo lambda de Church) y si tales ordenadores podrían resolver algún problema matemático.

### 3.5. Teoría descriptiva de la complejidad

En 1969 Fagin<sup>8</sup> decidió estudiar espectros (un espectro de una frase de primer orden es el conjunto de cardinalidades de sus modelos finitos) y el problema de Asser (1955): «¿Se cierra la clase de espectros al complementarse?». En 1970 sus investigaciones se ampliaron a los espectros generalizados (es decir, espectros existenciales de segundo orden en los que no se han cuantificado todos los símbolos de relación). Probablemente, el resultado más importante de Fagin fue su caracterización de los NP como la clase de espectros generalizados (1974). Actualmente el interés en este tema ha crecido exponencialmente, debido sobre todo al vínculo estrecho (que Fagin fue el primero en insinuar) entre la teoría de modelos finitos y la teoría de la complejidad.<sup>9</sup> De hecho, hay un

área consolidada dentro de la teoría de modelos finitos que trata explícitamente esta relación, y es la teoría descriptiva de la complejidad.

### 3.6. Entropía de Shannon

La entropía de Shannon<sup>10</sup> puede entenderse como la capacidad de captar un mensaje que se transmite a través de un canal dada la variedad de mensajes posibles. El planteamiento es que cuanto más cuesta entenderlo, más información proporciona el mensaje. Esa idea no se planteó como una medida de complejidad pero los autores posteriores la han utilizado como tal.

### 3.7. Complejidad de Goodman

Goodman<sup>11</sup> ha diseñado una categorización detallada de predicados extralógicos basados en la expresividad. Por ejemplo, se considera que un predicado general es más complejo que uno simétrico, ya que incluye a éste último como ejemplo específico. Del mismo modo, un predicado triádico es más complejo que uno diádico. Goodman desarrolla este punto de partida. En principio, al encontrarse ante dos teorías que se sustentan sobre el mismo número de pruebas experimentales, habría que elegir la más sencilla usando esta medida. La complejidad de una frase compleja es sencillamente la suma de las complejidades de los predicados que la componen, sea cual sea la estructura de la frase.

### 3.8. Complejidad de Kemeny

En el campo de la «simplicidad», Kemeny<sup>12</sup> atribuye complejidad integral a tipos de predicados extralógicos. Lo hace basándose en el algoritmo del número de modelos finitos no isomórficos que posee un predicado. Y partiendo de esta idea otorga a los predicados extralógicos una complejidad que podría emplearse para decidirse entre diversas teorías sustentadas por igual. Posee un estilo y una orientación similares a los del indicador de Goodman de la sección 3.7.

5. SMITH, C.U.M. (1990). *Elements of Molecular Neurobiology*. Chichester: Wiley.

6. CRUTCHFIELD, J.P. (1994). «The Calculi of Emergence: Computation, Dynamics and Induction». *Physica D*. N.º 75, págs.11-54.

7. VON NEUMANN, J. (1956). «The general and logical theory of automata». J.R. Newman (editor). *The World of Mathematics*. Nueva York: Simon and Schuster. Vol. 4, págs. 2070-2098; PAPANIMITRIOU, C. H. (1994). *Computational Complexity*. Reading, MA: Addison Wesley; FAGIN, R. (1973). «Contributions to the Model Theory of Finite Structures». Tesis doctoral presentada en la Universidad de California en Berkeley.

8. *Ibid.*, R. Fagin.

9. BANZHAF, W. (1994). «Self-Replicating Sequences of Binary Numbers: The Build-up of Complexity». *Complex Systems*. N.º 8, págs. 215-225.

10. SHANNON, C. E.; WEAVER. (1949). *The Mathematical Theory of Communication*. Urbana, IL: University of Illinois Press.

11. GOODMAN, N. (1966). *The Structure of Appearance*. Indianapolis: Bobbs-Merrill.

12. KEMENY, J.G. (1953). «Two Measures of Complexity». *The Journal of Philosophy*. N.º 52, págs. 722-733.

### 3.9. Complejidad de Horn y complejidad de redes

La complejidad de Horn de una función proposicional es la extensión mínima de una fórmula de Horn (en sus variables operativas) que define tal función. Esta complejidad fue definida por Aanderaa y Börger<sup>13</sup> como medida de la complejidad lógica de las funciones booleanas. Se vincula polinómicamente a la complejidad de redes o circuitos, que es el número mínimo de puertas lógicas necesarias para implementar una función lógica.<sup>14</sup>

### 3.10. Complejidad de la medida efectiva (EMC)

Grassberger<sup>15</sup> define la «complejidad de la medida efectiva» (EMC) de una pauta como el comportamiento asintótico de la cantidad de información exigida para predecir el siguiente símbolo en términos de granularidad. La EMC puede entenderse como la dificultad de predecir los valores futuros de una serie estacionaria, al medirse por el tamaño de expresión regular del modelo exigido. Badii y Politi<sup>16</sup> adoptan un enfoque similar.

### 3.11. Número de descripciones no equivalentes

Si un sistema puede modelarse de muchos modos distintos e irreconciliables, entonces siempre tendremos que aceptar un modelo incompleto de tal sistema. En tales circunstancias, es posible que el sistema muestre un comportamiento que sólo prediría otro modelo, por lo que tales sistemas son, de un modo fundamental, irreductibles. En consecuencia, Rosen<sup>17</sup> y Pattee<sup>18</sup> consideraron que la presencia de múltiples modelos no equivalentes era la característica clave de la complejidad. Casti<sup>19</sup> amplía este enfoque y define la complejidad como el número de descripciones no equivalentes que puede generar un observador de un sistema con el que interactúa. El observador debe elegir una familia de descripciones del sistema y una relación de equivalencia entre ellas, de manera que la complejidad es el número de clases de equivalencia en las que se divide la familia, dada la relación de equivalencia.

## 4. Propiedades de sistemas complejos

La búsqueda de las propiedades de los sistemas complejos está intrínsecamente ligada a la definición de complejidad. Diversos investigadores han emprendido la tarea de definir tales propiedades. Como ocurre en las definiciones de complejidad (sección 3), no existe una «lista» común de propiedades capaz de describirlas por completo.

### 4.1. Variedad

En un sistema complejo es probable que haya comportamientos y propiedades más variados. Por lo tanto, la variedad es un indicador de complejidad (aunque no siempre, ya que en ocasiones se hace necesario un sistema muy complejo para mantener el equilibrio). La variedad puede medirse con un simple recuento de los tipos, la extensión de valores numéricos o la mera presencia de cambios repentinos.

### 4.2. Dependencia

Heylighen<sup>20</sup> sugiere que la complejidad se incrementa cuando la variedad (distinción) y la dependencia (conexión) de partes o aspectos aumentan, y lo hacen en varias dimensiones. Entre ellas se incluye al menos las tres dimensiones espaciales ordinarias, la estructura geométrica, la dimensión de escala espacial, la dimensión de tiempo o dinámica, y la dimensión de escala temporal o dinámica. Para mostrar que la complejidad ha aumentado en conjunto, basta con mostrar que –si todo es igual– la variedad y/o la conexión han aumentado en al menos una dimensión.

### 4.3. Irreductibilidad

La irreductibilidad es una fuente de complejidad. Nelson<sup>21</sup> afirma que la irreductibilidad es un factor clave en sistemas complejos, y también se encuentra un enfoque similar en los escritos de Anderson,<sup>22</sup> que

13. AANDERAA, S.O.; BRÖGER E. (1981). «The equivalence of Horn and network complexity of Boolean functions». *Acta Informatica*. N.º 15, págs. 303-307.

14. SAVAGE, J. E. (1987). *The Complexity of Computing*. Melbourne, FL: Krieger Publishing Company.

15. GRASSBERGER, P. (1986). «Towards a Quantitative Theory of Self-Generated Complexity». *International Journal of Theoretical Physics*. N.º 25, págs. 907-938.

16. BADI, R.; POLITI, A. *Complexity: hierarchical structures and scaling in physics*. Cambridge: Cambridge University Press.

17. ROSEN, R. (1977). «Complexity and system descriptions». En: W.E. HARNETT (editor). *Systems: Approaches, theories, Applications*. Dordrecht, Holanda: D. Reidel. Págs. 169-175.

18. PATTEE, H. (1977). «Dynamic and Linguistic Modes of Complex Systems». *International Journal of General Systems*. N.º 3, págs. 259-266.

19. CASTI, J.L. (1986). «On System Complexity: Identification, Measurement and Management». En: J. CASTI; A. KARLQUIST (editores). *Complexity Language and Life: Mathematical Approaches*. Berlín: Springer. Págs. 146-173.

20. HEYLIGHEN, F. (1996). «The Growth of Structural and Functional Complexity during Evolution». En: F. HEYLIGHEN; D. AERTS. *The Evolution of Complexity*. Dordrecht, Holanda: Kluwer Academic Publishers.

21. NELSON, R.J. (1976). «Structure of Complex Systems». *Philosophy of Science Association*. N.º 2, págs. 523-542.

22. ANDERSON, P.W. (1972). «More is Different». *Science*. N.º 177, págs. 393-396.

señala la importancia del tamaño para el comportamiento cualitativo, y Wimsatt,<sup>23</sup> quien afirma que la evolución de múltiples funciones superpuestas limitará la simplificación en biología.

#### 4.4. Capacidad de sorprender

Los sistemas muy simples y por lo tanto bien entendidos no poseen la capacidad de sorprender, por lo que se considera una propiedad esencial de los sistemas complejos.<sup>24</sup>

#### 4.5. Ruptura de la simetría

Heylighen<sup>25</sup> afirma que entonces la complejidad puede caracterizarse por la falta de simetría o «ruptura de simetría», por el hecho de que ninguna parte o aspecto de una entidad compleja puede proporcionar información suficiente para predecir real o estadísticamente las propiedades de las demás partes. Esta idea vuelve a enlazar con la dificultad de modelado asociada a sistemas complejos.

#### 4.6. Complejidad como elemento relativo al marco de referencia

Edmonds<sup>26</sup> destaca que la complejidad depende necesariamente del lenguaje utilizado para modelar un sistema. Afirma que la complejidad efectiva depende del marco elegido a partir del cual se contempla/modela el sistema de estudio. La importancia de la escala para modelar fenómenos también conduce a Badii y Politi<sup>27</sup> a centrar su caracterización de la complejidad basándose solamente en tales efectos jerárquicos y de escala.

#### 4.7. Punto medio entre orden y desorden

En ocasiones la complejidad se sitúa en un punto medio entre el orden y el desorden. Edmonds<sup>28</sup> señala que la definición de complejidad como punto medio entre orden y desorden depende del nivel de representación: lo que parece complejo en una representación

puede parecer ordenado o desordenado en una representación a una escala distinta.

#### 4.8. Complejidad a través del cambio de estado

La investigación de sistemas complejos denomina cambio de estado a la transición entre las zonas de pautas de actividad sencilla y pautas de actividad compleja. Kauffman<sup>29</sup> ha modelado un hipotético sistema de circuitos de moléculas que pueden apagarse o encenderse las unas a las otras para catalizar o inhibir uno de sus productos. Como consecuencia de esta catálisis o cierre colectivo e interconectado, se catalizan más moléculas complejas, que vuelven a ejercer de catalizadores de moléculas aún más complejas. Kauffman afirma que la vida puede darse en el propio cierre catalítico, siempre y cuando se alcance una diversidad molecular significativa. El estado de suspenso entre la estabilidad y la flexibilidad suele denominarse «el borde del caos».

#### 4.9. La vida al borde del caos

Langton y Packard<sup>30</sup> observaron el comportamiento de autómatas celulares. Descubrieron que, aunque los autómatas celulares obedecen reglas simples de interacción del tipo descrito por Wolfram,<sup>31</sup> pueden desarrollar pautas complejas de actividad. Como estas pautas dinámicas y complejas se desarrollan y recorren el sistema entero, surgen estructuras locales a partir de reglas de actividad local, lo que es un rasgo típico de los sistemas complejos. Langton y Packard plantearon la hipótesis de que el tercer estadio de comunicación elevada es también el mejor lugar para la adaptación y el cambio, y de hecho sería el mejor lugar para proporcionar el mayor número de oportunidades para que el sistema desarrollara estrategias dinámicas de supervivencia. Además sugieren que este estadio atrae a sistemas en evolución. Posteriormente, bautizaron a la fase de transición de este tercer estadio como «la vida al borde del caos».<sup>32</sup>

23. WIMSATT, W. (1972). «Complexity and Organisation». En: K. SCHAEFFNER; R. COHEN (editores). *Studies in the Philosophy of Sciences*. Dordrecht, Holanda: Reidel. Págs. 67-86.

24. *Ibid.*, B. Edmonds.

25. *Ibid.*, R.J. Nelson.

26. *Ibid.*, B. Edmonds.

27. *Ibid.*, R. Badii, A. Politi.

28. *Ibid.*, Edmonds.

29. KAUFFMAN, S. (1995). *At Home in the Universe. The Search for Laws of Complexity*. Nueva York: Oxford University Press. Págs. 23-112.

30. LANGTON, C. (1992). «Life at the edge of chaos». En: C. LANGTON [et al.]. *Artificial Life II: Santa Fe Institute Studies in the Sciences of Complexity*. Reading, MA: Addison-Wesley. Vol. 10, págs. 41-91.

31. WOLFRAM, S. (1986). *Theory and Applications of Cellular Automata*. Singapur: World Scientific Press.

32. *Ibid.*, C. Langton.

## 5. Aplicación de principios de sistemas complejos a la creación de obras de arte interactivas y generadas por ordenador en internet

Para resumir, podemos ver que aunque hay varias definiciones y ejemplos distintos de sistemas complejos (Kolmogorov-Chaitin-Solomonoff, Hinegardner y Engelberg, Crutchfield, Papadimitriou, Grassberger, Badii y Politi, Pattee, Casti, Heylighen y la perspectiva general y completa de Edmonds), de hecho no existe una teoría de sistemas complejos unificada, ni un «manual» como tal sobre cómo crear sistemas complejos. Por otra parte, los modelos de Kauffman y Langton y Packard sugieren sistemas complejos adaptativos, sistemas en el «borde del caos» en los que los cambios internos pueden describirse siguiendo una distribución de ley potencial. Estos sistemas se encuentran en el punto de máxima aptitud computacional, máxima adecuación y máxima capacidad de evolucionar.

Como ambicionábamos crear una obra compleja e interactiva que pudiera cambiar constantemente, adaptarse y evolucionar cuando los usuarios interactuaran con el sistema, los enfoques de Kauffman<sup>33</sup> y Langton y Packard<sup>34</sup> con relación a la complejidad, es decir, los que se centraban en la complejidad evolutiva, nos parecieron los más adecuados para nuestros objetivos.

## 6. Life Species y Life Species II: modelar complejidad para arte interactivo

Life Species y Life Species II se basan en nuestro objetivo de crear una obra de arte interactiva que muestre algunos de los rasgos de los sistemas complejos.

### 6.1. Life Species

Life Species fue encargada por el ICC-NTT InterCommunication Museum de Tokio, donde se presentó la primera versión en la primavera de 1997.<sup>35</sup> A través de una página web especial de Life Species, gente de todo el mundo podía interactuar con el sistema: los visitantes podían crear su propia criatura artificial tan sólo escribiendo un mensaje de correo y enviándolo a la página web de Life Species (figura 1). Entonces la criatura «nacía» en el entorno tridimensional interactivo de Life Species, en el interior del museo ICC-NTT.

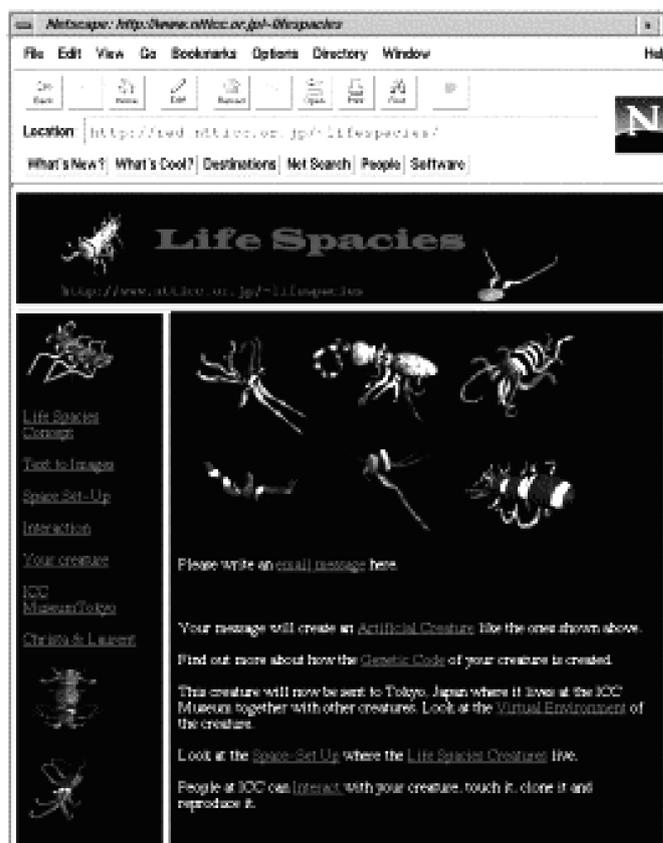


Figura 1. Página web de Life Species

En el museo de Tokio, el sistema interactivo consistía en dos sitios de interacción independientes unidos mediante una línea de datos, lo que permitía a los visitantes en ubicaciones remotas mostrarse e interactuar dentro del mismo espacio virtual tridimensional. Como en la instalación interactiva de telepresencia Time\_Lapse, los visitantes del museo se introducían en un complejo mundo de paisajes tridimensionales en pleno desarrollo que reaccionaba ante los movimientos corporales y los gestos de los visitantes, como se muestra en la figura 2.

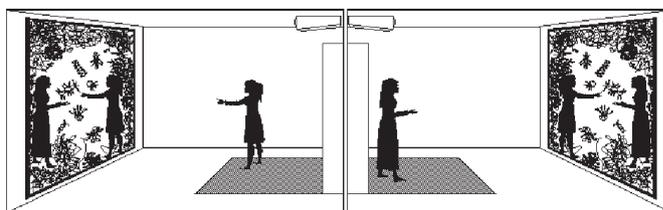


Figura 2. Sistema interactivo Life Species en el museo de Tokio

33. *Ibid.*, S. Kauffman.

34. *Ibid.*, C. Langton.

35. SOMMERER, C.; MIGNONNEAU, L. (1997). «Life Species». En: *ICC Concept Book*. Tokio: NTT-ICC Tokyo. Págs. 96 -101.

Los dos sitios de interacción estaban formados por un suelo blanco de 4 x 2,1 metros situado delante de una caja de luz y una pantalla de proyección de 4 x 3 metros. Como en Time\_Lapse, se utilizó una técnica de luminancia para extraer la imagen y el contorno de cada usuario desde el fondo. A continuación estas imágenes se introducían en el entorno tridimensional virtual. Como consecuencia de este proceso de introducción e integración, cada usuario se veía a sí mismo y al otro usuario en tres dimensiones dentro de un mundo de plantas artificiales. Entonces los visitantes del museo podían interactuar los unos con los otros, provocando el crecimiento de plantas tridimensionales a través de sus gestos, y encontrarse con las criaturas tridimensionales procedentes de la página web de Life Species. Cuando los usuarios del museo tocaban una criatura, la interacción provocaba que creara un clon de sí misma. Y si usuarios de uno y otro lado atrapaban a una criatura con las manos, podían hacer que se aparearan y crearan descendencia a través del intercambio genético de los códigos de sus progenitores.

### 6.1.1. Editor de texto a formas de Life Species

Para diseñar las criaturas, desarrollamos un editor especial de texto a formas que permite traducir el texto escrito de los mensajes de correo entrantes al código genético de una criatura. El editor de texto a formas se basó en la idea de vincular los caracteres y la sintaxis de un texto a parámetros específicos en el diseño de la criatura.

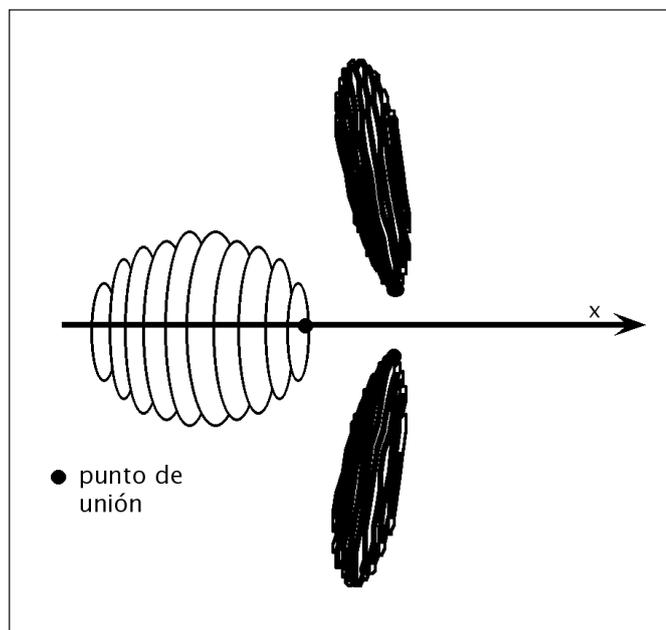


Figura 3. Criatura de Life Species con dos cuerpos y un par de extremidades

El cuerpo de una criatura se compone por defecto de una esfera consistente en 100 vértices, es decir, 10 anillos con 10 vértices cada uno. Todos los vértices pueden modificarse en los ejes X, Y

y Z, de modo que la esfera se estira y crea nuevas figuras. Varias figuras pueden unirse entre ellas siempre y cuando su punto de unión se encuentre en el eje X. Si el punto de unión no es el eje X, se crea una extremidad en vez de una figura del cuerpo. Luego este miembro se copia y pega en una posición simétrica a la original. La figura 3 muestra una criatura con dos esferas por cuerpo y un par de extremidades.

Los parámetros X, Y y Z de cada uno de estos 100 vértices dependen de la secuencia de caracteres del texto: se pueden estirar y escalar, los valores de color y textura en cada cuerpo y extremidad pueden modificarse, se puede alterar el número de figuras corporales y extremidades y crear nuevas ubicaciones para los puntos de unión de cuerpos y extremidades. Dado que cada uno de los parámetros de los vértices es modificable y todos los cuerpos y extremidades también pueden alterarse, se dispone de unas 50 funciones diferentes para diseñar la criatura. Las funciones de diseño se subsumen en una tabla de diseño de funciones (figura 4).

- función1 estirar cuerpo/extremidades por defecto en x
- función2 estirar cuerpo/extremidades por defecto en y
- función3 estirar cuerpo/extremidades por defecto en z
- función4 establecer la siguiente función de extensión en global
- función5 establecer la siguiente función de extensión en un vértice
- función6 establecer la siguiente función de extensión en un anillo
- función7 crear una nueva ubicación para un punto de unión
- función8 copiar una nueva ubicación para un punto de unión
- función9 componer una nueva textura para cuerpo/extremidades
- función10 copiar textura de cuerpo/extremidades
- función11 cambiar parámetros de ROJO en textura de cuerpo/extremidades
- función12 cambiar parámetros de VERDE en textura de cuerpo/extremidades
- función13 cambiar parámetros de AZUL en textura de cuerpo/extremidades
- función14 cambiar patrones de textura de cuerpo/extremidades
- función15 intercambiar posiciones de cuerpos/extremidades
- función16 copiar cuerpo/extremidades
- función17 crear un nuevo cuerpo/extremidades
- función18 añadir o sustituir alguna de las funciones anteriores
- función19 aleatorizar los siguientes parámetros
- función20 copiar partes de la operación anterior
- función21 modificar duración de la vida (24 horas por defecto)
- función22 añadir el nuevo parámetro al parámetro previo
- función23 ignorar el parámetro actual
- función24 ignorar el siguiente parámetro
- función25 sustituir el parámetro anterior por un nuevo parámetro
- .....
- función50

Figura 4. Tabla de diseño de funciones de Life Species

A continuación, para traducir los caracteres del mensaje de texto a estos valores de diseño de funciones, asignamos primero un valor ASCII a cada carácter, para lo cual se sigue la tabla estándar ASCII que se muestra en la figura 5.

33!	34"	35#	36\$	37%	38&	39'	
40(	41)	42*	43+	44,	45-	46.	47/
480	491	502	513	524	535	546	557
568	579	58:	59;	60<	61=	62>	63?
64@	65A	66B	67C	68D	69E	70F	71G
72H	73I	74J	75K	76L	77M	78N	79O
80P	81Q	82R	83S	84T	85U	86V	87W
88X	89Y	90Z	91[	92\	93]	94^	95_
96`	97a	98b	99c	100d	101e	102f	103g
104h	105i	106j	107k	108l	109m	110n	111o
112p	113q	114r	115s	116t	117u	118v	119w
120x	121y	122z	123{	124	125}	126~	

Figura 5. Tabla ASCII

Cada carácter remite a un número entero. Continuamos asignando este valor a una función aleatoria *rseed*. En nuestro texto de ejemplo de la figura 4, la *T* de *This* posee el valor ASCII 84, así que la función aleatoria de la *T* se convierte en *rseed(84)*. Esta función aleatoria define una secuencia infinita de números aleatorios distribuidos linealmente con una precisión de coma flotante de 4 bytes (los valores flotantes se hallan entre 0,0 y 1,0).

Estos números aleatorios para el primer carácter de la palabra *This* se convertirán en los valores utilizados para los parámetros de modificación en la tabla de diseño de funciones. Hay que destacar que el número aleatorio que empleamos es lo que se denominaría un «pseudoaleatorio», generado por un algoritmo con precisión de 48 bits, lo que significa que, si se vuelve a llamar a la misma función *rseed*, se generará la misma secuencia de números aleatorios distribuidos linealmente. La determinación de las funciones de diseño en la tabla de diseño de funciones que se actualicen realmente dependerá de los caracteres posteriores del texto, en este caso *his*. Luego les asignamos valores ASCII (104 a la *h*, 105 a la *i*, 115 a la *s*), que volverán a proporcionarnos las funciones aleatorias *rseed(104)*, *rseed(105)* y *rseed(115)*.

A continuación estas funciones aleatorias se utilizan para actualizar y modificar las funciones de diseño correspondientes en la tabla de diseño de funciones de consulta, entre la función de diseño 1 y la función 50. Por ejemplo, al multiplicar el primer número aleatorio de *rseed(104)* por 10 obtenemos el número entero que asigna la cantidad de funciones que se actualizarán: de las 50 que hay, las elegidas para actualizarse se decidirán a partir de los siguientes números aleatorios de *rseed(104)* (como disponemos de 50 funciones distintas, los si-

guientes tipos reales se multiplican por 50 para crear números enteros). La figura 5 muestra en detalle cómo funciona la asignación entera de números aleatorios a funciones de diseño. Como se ha mencionado antes, los valores de tipo real para los parámetros de actualización proceden de la función aleatoria del primer carácter de la palabra, *rseed(84)*. Este procedimiento se ilustra íntegramente en la figura 6.

Palabra de ejemplo: <i>This</i>
$T \Rightarrow rseed(84) \Rightarrow \{0,36784, 0,553688, 0,100701, \dots\}$ (valores reales para los parámetros de actualización)
$h \Rightarrow rseed(104) \Rightarrow \{0,52244, 0,67612, 0,90101, \dots\}$ # $0,52244 * 10 \Rightarrow$ obtener número entero 5 $\Rightarrow$ se llama a 5 funciones distintas en la tabla de diseño de funciones
# $0,67612 * 50 \Rightarrow$ obtener número entero 33 $\Rightarrow$ la función 33 en la tabla de diseño de funciones se actualizará mediante el valor 0,36784 del primer valor de <i>rseed(84)</i>
# $0,90101 * 50 \Rightarrow$ obtener número entero 45 $\Rightarrow$ la función 45 en la tabla del diseño de funciones se actualizará mediante el valor 0,553688 del segundo valor <i>rseed(84)</i> hasta el quinto valor

Figura 6. Ejemplo de asignación de funciones aleatorias y funciones de diseño

Como se ha explicado anteriormente, el «módulo» básico del cuerpo de la criatura es una esfera de color blanco por defecto y sin textura. Cuando se envían mensajes, el texto entrante modifica y «esculpe» este módulo inicial cambiando su forma, tamaño, color, textura, número de cuerpos/extremidades, copiando partes y demás. Dependiendo de la complejidad del texto, el cuerpo y las extremidades de la criatura quedarán cada vez más definidas, moduladas y variadas. Como suele haber mucha variación entre los textos enviados por distintas personas, las criaturas de por sí ya difieren mucho en su

Date: Sun, 01 Nov 1998 13:14:32 +0900
From: Christa Sommerer <christa@mic.atr.co.jp>
To: life@lc.nttcc.or.jp
CC: christa@mic.atr.co.jp
Subject: test creature1
This is a test creature.

Figura 7. Mensaje de correo enviado al entorno Life Species

aspecto, lo que supone una criatura única por autor. La figura 7 muestra un ejemplo de mensaje de correo corto y sencillo enviado a la página web de Life Spacies.

### 6.1.2. Imagen de la criatura de Life Spacies

Cuando el servidor de Tokio recibió el mensaje, la criatura comenzó a vivir en el entorno virtual, y a cambio el autor del texto recibió una imagen de su criatura. La figura 8 muestra una imagen de la criatura creada mediante el mensaje de texto de la figura 7. Como el mensaje de texto era bastante corto, la criatura correspondiente está formada por un solo cuerpo y un par de extremidades, por lo que se asemeja a la figura por defecto pero tiene extremidades más largas y el cuerpo en forma de corazón.

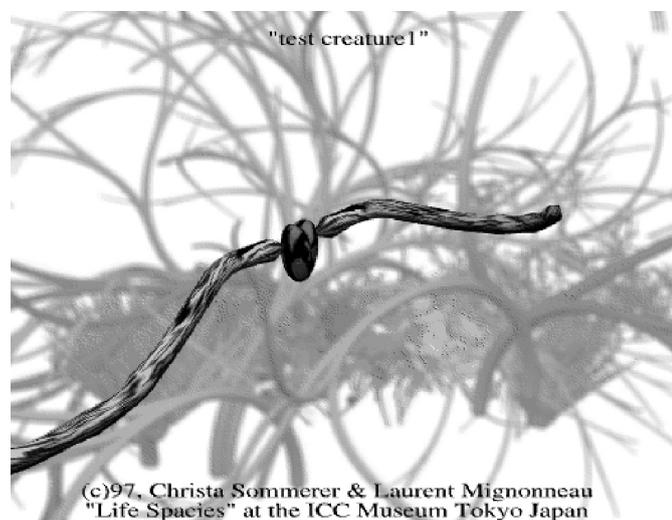


Figura 8. La criatura de Life Spacies creada con el mensaje de correo de la figura 7

### 6.1.3. Variaciones en el diseño de la criatura

Los mensajes complejos con más caracteres, palabras y sintaxis variada generan criaturas más elaboradas con más figuras corporales,

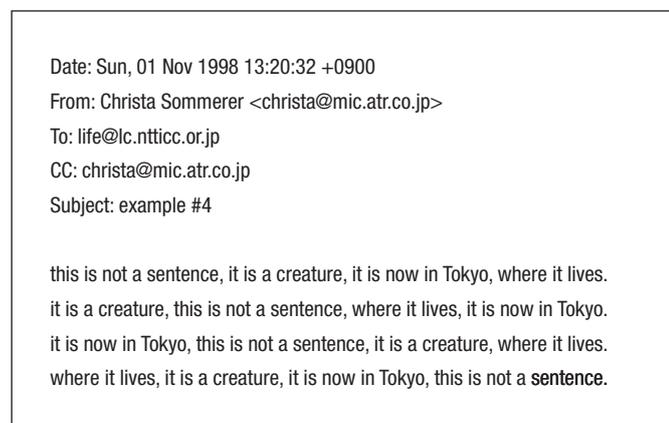


Figura 9. Un mensaje de correo más complejo

extremidades y variaciones de forma, textura, tamaño y color. El mensaje de texto de la figura 9 y la criatura correspondiente de la figura 10 muestran este incremento de la complejidad.

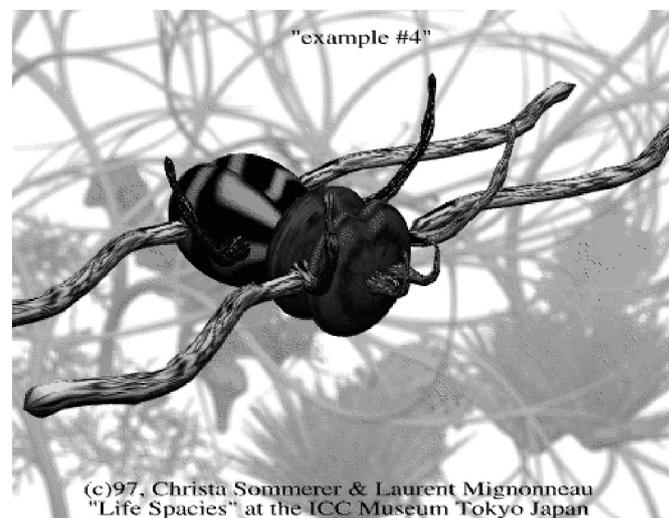


Figura 10. Criatura creada mediante un mensaje de correo más complejo

## 6.2. Life Spacies II

En 1999 creamos una versión autónoma de Life Spacies que era más transportable y podía mostrarse en exposiciones itinerantes. La integración tridimensional de visitantes en el entorno virtual se omitió en esta versión. El sistema de Life Spacies II presentaba una interfaz gráfica de usuario (GUI) en un ordenador portátil colocado sobre un podio con una pantalla de proyección de 4 x 3 metros.

### 6.2.1. Life Spacies II – GUI para crear y alimentar criaturas

En Life Spacies II empleamos una GUI para que los usuarios pudieran escribir mensajes de texto desde ubicaciones remotas (figura 11) o directamente en la propia instalación. Como ocurría en Life Spacies, el texto escrito se utilizaba como código genético para crear formas tridimensionales. Cuando se escribe un texto en la ventana izquierda de la GUI, la criatura correspondiente aparece de inmediato en la ventana derecha de la GUI, así como en la gran pantalla de proyección de la instalación.

Además de crear criaturas, los usuarios también pueden «alimentarlas» añadiendo caracteres textuales en la ventana derecha de la GUI. El usuario puede decidir la cantidad y el tipo de texto y dónde desea colocarlo tecleando caracteres textuales concretos en la página web. Las criaturas recogen el texto (o comida), que se muestra simultáneamente en la pantalla de proyección.

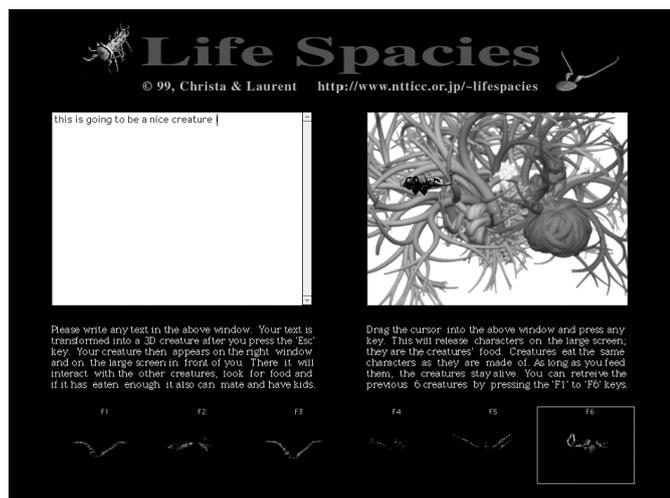


Figura 11. Interfaz gráfica de usuario de Life Species II

### 6.2.2. Life Species II – Editor de texto a formas

El editor de texto a formas de Life Species II es el mismo de Life Species (véase el apartado 6.1.1).

## 6.3. Life Species y Life Species II – Comportamiento de las criaturas

### 6.3.1. Energía y velocidad

El comportamiento de una criatura depende básicamente de dos parámetros: 1) su nivel de energía (E) y b) su velocidad (V) o capacidad para moverse. Mientras que el nivel de energía es un valor que cambia constantemente cuando la criatura se desplaza en su entorno y disminuye al aumentar el movimiento, el valor de la velocidad viene dado por la tipología corporal de la criatura.

**Velocidad (V):**

- Depende del tamaño del cuerpo y de las extremidades de la criatura.
- Decide la velocidad a la que se mueve la criatura.

**Energía (E):**

- E = 1 en el momento del nacimiento.
- El movimiento reduce E.
- Si  $E < 1$ , la criatura tiene hambre.
- Si  $E > 1$ , la criatura puede aparearse.

Figura 12. Parámetros de decisión del comportamiento de la criatura

La que tenga el cuerpo grande y las extremidades pequeñas tenderá a moverse más lentamente que una criatura con el cuerpo pequeño y las extremidades largas. Además, la forma del cuerpo y las extremidades de la criatura influirán en su capacidad para desplazarse. Por otra parte, el valor de la velocidad se establece al crearse la criatura mediante la disposición de los caracteres de texto en su código genético, que se interpreta y traduce a la tabla de diseño de funciones tal y como se explica en el apartado 6.1.1 y en la bibliografía.<sup>36</sup>

### 6.3.2. Interacción entre criaturas

La interacción de la criatura con otras criaturas depende de cuánta energía (E) posea en un momento determinado y de la velocidad (V) a la que pueda desplazarse en el entorno. Si, por ejemplo, el nivel de energía de la criatura alcanza un determinado valor  $E < 1$ , la criatura pasa hambre y quiere comer. Por otra parte, si el nivel de energía asciende a  $E > 1$ , la criatura quiere aparearse con otras criaturas. Las figuras 12 y 13 muestran esta relación entre niveles de energía y comportamiento en la alimentación y el apareamiento.

**Alimentación:** Si  $E < 1$ , la criatura quiere comer caracteres textuales y se come los mismos caracteres de su código genético (la criatura "John" se come: "J", "o", "h", "n").

**Apareamiento:** Si  $E > 1$ , la criatura quiere aparearse; si tiene éxito, los progenitores intercambiarán su código genético y puede nacer un bebé criatura.

Figura 13. Parámetros de interacción de criaturas

### 6.3.3. Alimentación

Una criatura cuyo nivel de energía haya descendido a  $E < 1$  pasa «hambre» y desea comer caracteres textuales proporcionados por el usuario a través del editor de entrada textual de Life Species y Life Species II. El tipo de caracteres que aparezcan y cuándo lo harán dependerá solamente de las decisiones del usuario. Las criaturas también prefieren ciertos tipos de comida y sólo comen caracteres textuales que formen parte de su código genético. Por ejemplo, una criatura cuyo código genético es *John* sólo comerá caracteres *J, o, h* y *n*. Al comer caracteres textuales, la criatura acumula más energía y su nivel de energía acaba ascendiendo a  $E > 1$ . No obstante, puede que la criatura necesite comer unos cuantos caracteres, ya que las criaturas se desplazan enérgicamente cuando buscan caracteres textuales. En general, las criaturas más veloces consiguen atrapar

36. *Ibid.*

la comida con mayor rapidez, pero el usuario puede «ayudar» a las criaturas más lentas colocándoles la comida delante. En este caso, incluso una criatura más lenta puede llegar a conseguir comida suficiente para alcanzar el nivel E>1.

### 6.3.4. Apareamiento

Cuando una criatura logra acumular energía hasta alcanzar el nivel E>1, se convierte en compañero potencial para el apareamiento. Entonces se pone a buscar a un compañero apto, cuyo nivel de energía también esté por encima de 1: así que los dos progenitores potenciales de la criatura se desplazarán el uno hacia el otro e intentarán chocar. Si tienen éxito, los dos progenitores intercambian su código genético a través de una operación de cruce, como resultado de la cual nace una criatura bebé. Esta cría lleva el código genético de sus progenitores con algunas mutaciones. La figura 14 muestra un ejemplo del proceso de apareamiento.

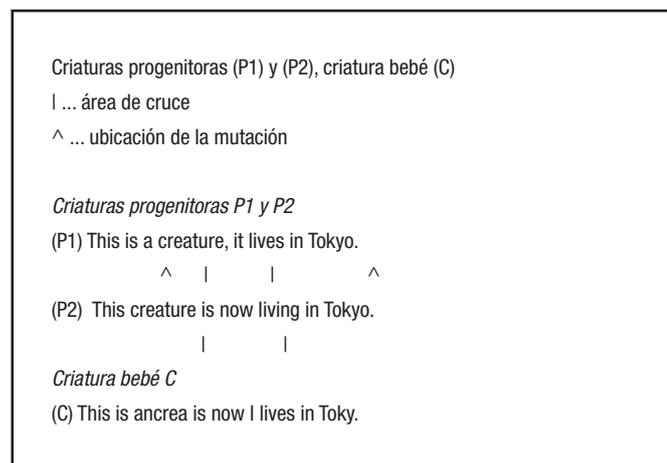


Figura 14. Proceso de apareamiento y nacimiento de la criatura bebé

### 6.3.5. Crecimiento y muerte

La duración de la vida de una criatura no está predeterminada, sino que se ve influida por cuánto come. A través del alimento la criatura aumenta el tamaño corporal hasta alcanzar un máximo de unas cuatro veces su tamaño original. Por otra parte, una criatura pasará hambre cuando no coma suficientes caracteres textuales y acabará muriendo y hundiéndose en el suelo.

## 6.4. Life Spacies y Life Spacies II – Evaluación de la complejidad

El movimiento constante, la alimentación, el apareamiento y la reproducción de las criaturas acaban generando un sistema complejo de interacciones que muestra rasgos de evolución artificial. Aunque el parámetro de selección consiste en atrapar la comida rápidamente y así aparearse con mayor frecuencia, los usuarios pueden invertir este proceso «ayudando» a criaturas más lentas a adquirir energía suficiente para poder aparearse también. Las decisiones de los usuarios sobre cómo escribir los mensajes de texto y sobre cómo y dónde alimenten a las criaturas añaden así cambios constantes y crean un sistema que presenta interacciones complejas entre criaturas así como entre usuarios y criaturas. Las criaturas se aparean y alimentan de caracteres textuales. Cuando volvemos a las definiciones de sistemas complejos del apartado 3, vemos que Life Spacies II muestra los siguientes rasgos asociados con sistemas complejos: adaptarse y organizarse, mutar y evolucionar, expandir su diversidad, reaccionar ante los vecinos y el control externo, explorar sus opciones y reproducirse.

## 7. Conclusiones

Inspirados por la idea de aplicar algunos rasgos de sistemas complejos a obras de arte interactivas complejas y adaptativas, diseñamos los sistemas interactivos Life Spacies y Life Spacies II. Aunque desde mediados de los noventa diversos artistas y diseñadores han creado obras de arte o programas de entretenimiento basados en la vida artificial, la mayoría de estos productos proporcionan a los usuarios criaturas predefinidas o partes de criaturas que los usuarios han de elegir o ensamblar.<sup>37</sup> Dado que este tipo de estructuras permiten decidir muy pocos elementos a los usuarios de los sistemas, diseñamos Life Spacies con un sistema más flexible que permite a los usuarios tomar más decisiones en los ámbitos del diseño y la interacción. El texto escrito, aportado al azar por los usuarios del sistema, se utiliza como código genético, y nuestro editor de texto a formas traduce los textos escritos a criaturas autónomas tridimensionales cuyos cuerpos, comportamientos, interacciones y supervivencia se basan exclusivamente en su código genético y en las interacciones de los usuarios. Al interactuar con estos sistemas, los usuarios los vuelven cada vez más complejos y les otorgan rasgos propios de sistemas complejos como variedad y dependencia, irreductibilidad, ruptura de la simetría, adaptación y organización, mutación y evolución, expansión de la diversidad, reacción ante los vecinos y el control externo, exploración de sus opciones y reproducción.

37. GRAND, S.; CLIFF, D.; MALHOTRA, A. (1996). «Creatures: artificial life autonomous software agents for home entertainment». *Millennium Technical Report 9601*. University of Sussex Technical Report CSR434; HURRY, M.; PROPHET, J.; SELLEY, G. (2000). «TechnoSphere». En: *AlifeVII Artificial LifeVII in-house Conference Proceedings*. Portland, OR; VENTRELLA, Jeffrey. «Darwin Pond». <<http://www.ventrella.com/Darwin/darwin.html>>; Life Drop. <<http://www.virtual-worlds.net/lifedrop/>>; Sony Communication Network Corporation. «PostPet email software». <<http://www.sony.com.sg/>>

## Agradecimientos

Esta investigación se llevó a cabo en los ATR Media Integration and Communications Research Laboratories de Kioto, Japón. Damos especialmente la gracias a John Casti y Tom Ray por sus continuas y valiosas discusiones al respecto.

## Referencias

Una versión más corta de este texto se publicó por primera vez en SOMMERER, Christa; MIGNONNEAU, Laurent (2000). «Modeling Complex Systems for Interactive Art». En: S. HALLOY; T. WILLIAMS (editores). *Applied Complexity - From Neural Nets to Managed Landscapes*. Christchurch, Nueva Zelanda: Institute for Crop & Food Research. Págs. 25-38.

## Cita recomendada

---

SOMMERER, Christa; MIGNONNEAU, Laurent (2009). «Life Spacies y Life Spacies II: modelar sistemas complejos para arte interactivo». En: «Arte, cultura y ciencias de la complejidad» [nodo en línea]. *Artnodes*. N.º 9. UOC. [Fecha de consulta: dd/mm/aa].

<[http://artnodes.uoc.edu/ojs/index.php/artnodes/article/view/n9\\_sommerer\\_mignonneau/n9\\_sommerer\\_mignonneau](http://artnodes.uoc.edu/ojs/index.php/artnodes/article/view/n9_sommerer_mignonneau/n9_sommerer_mignonneau)>

ISSN 1695-5951

© 1997-1999 Christa Sommerer y Laurent Mignonneau

## CV

**Christa Sommerer y Laurent Mignonneau**

Departamento de Culturas de la Interfaz  
de la Universidad de Arte y Diseño de Linz  
[christa.sommerer@ufg.ac.at](mailto:christa.sommerer@ufg.ac.at)  
[laurent.mignonneau@ufg.ac.at](mailto:laurent.mignonneau@ufg.ac.at)



Christa Sommerer y Laurent Mignonneau son una pareja de artistas e investigadores de los medios reconocidos internacionalmente, cuya veintena de obras de arte interactivo puede consultarse en <http://www.interface.ufg.ac.at/christa-laurent>. Estas obras se han mostrado en más de 200 exposiciones y se encuentran en museos y colecciones de medios de todo el mundo, incluido el Van Gogh Museum de Amsterdam, el Museo de Ciencia e Industrias de Tokio, el Media Museum del ZKM en Karlsruhe, la Fondation Cartier de París, el Ars Electronica Center de Linz, el museo NTT-ICC de Tokio, el NTT Plan-Net de Nagoya, el Shiroishi Multimedia Art Center de Shiroishi, la House of Shiseido de Tokio y la fundación Itaú Cultural de São Paulo.

De las obras interactivas de Mignonneau y Sommerer se ha dicho que «marcan un hito» (Toshiharu Itoh, museo NTT-ICC de Tokio) por desarrollar interfaces naturales e intuitivas y porque a menudo aplican principios científicos de vida artificial, complejidad y sistemas interactivos a sus innovadores diseños de interfaces. Han ganado importantes premios internacionales de arte en medios, entre ellos el premio Golden Nica de Arte Interactivo en 1994 (Ars Electronica de Linz).

Mignonneau y Sommerer han escrito sobre vida artificial, complejidad, interactividad y diseño de interfaces y han impartido múltiples charlas y clases en universidades, conferencias internacionales y simposios. Durante diez años han trabajado de investigadores y profesores en los ATR Research Labs de Kioto y en IAMAS en Ogaki (ambos en Japón), y actualmente dirigen el Departamento de Culturas de la Interfaz en la Universidad de Arte y Diseño de Linz, Austria, que se especializa en arte interactivo, medios interactivos y diseño de interfaces. Han publicado varios libros: *Christa Sommerer & Laurent Mignonneau - Interactive Art Research* (2009). G. Stocker, C. Sommerer, L. Mignonneau (editores). Viena/Nueva York: Springer Verlag. ISBN: 978-3-211-99015-5; *The Art and Science of Interface and Interaction Design* (2008). C. Sommerer, L.C. Jain, L. Mignonneau (editores). Springer Verlag, vol. 1, xiv; *Studies in Computational Intelligence*, vol. 141, ISBN: 978-3-540-79869-9; *Interface Cultures - Artistic Aspects of Interaction* (2008). C. Sommerer, L. Mignonneau, D. King (editores). Transcript Verlag, ISBN: 978-3-89942-884-1; *Art @ Science* (1998). C. Sommerer, L. Mignonneau (editores). Viena/Nueva York: Springer Verlag.

[www.interface.ufg.ac.at/christa-laurent/](http://www.interface.ufg.ac.at/christa-laurent/)



Universitat Oberta  
de Catalunya

[www.uoc.edu](http://www.uoc.edu)