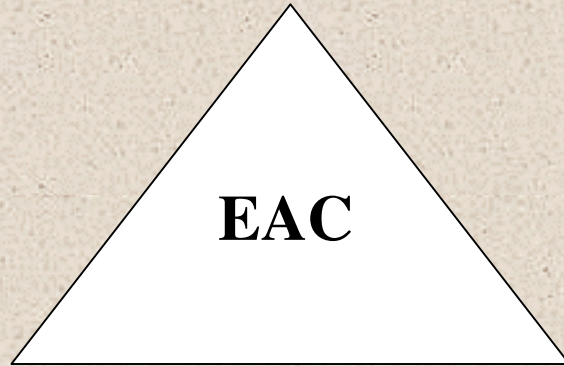


**Conciencia**



**Evolución**

**Afectividad**

<http://www.fgalindosoria.com/eac/>

# **EVOLUCIÓN Y SISTEMAS EVOLUTIVOS**

<http://www.fgalindosoria.com/eac/evolucion/>

La evolución, el crecimiento, el aprendizaje, la vida, el pensamiento, la transformación de nuestra imagen de la realidad, los procesos de descomposición, el desarrollo y transformación de las empresas, sociedades, organizaciones, países, galaxias, universos, etc., son manifestaciones de un mismo proceso general de transformación o cambio al que por facilidad llamamos evolución.

## **Fernando Galindo Soria**

[www.fgalindosoria.com](http://www.fgalindosoria.com)

[fgalindo@ipn.mx](mailto:fgalindo@ipn.mx)

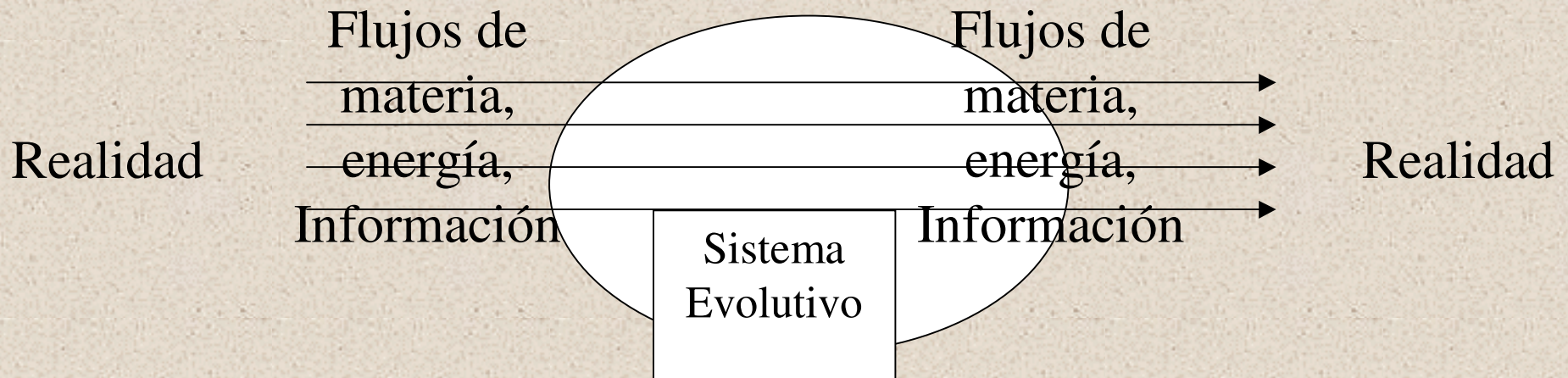
**21 de Septiembre del 2002, Octubre del 2009**

Se presenta un concepto general de los

***Sistemas Evolutivos,***

*como*

*sistemas capaces de transformarse permanentemente a partir de los flujos de materia, energía e información que los cruzan.*



Por los flujos de materia, energía, información el Sistema evoluciona

**Evolución 1995**

Primero se vera su relación con los sistemas formales y con los sistemas que aprenden.

Se mostraran algunos ejemplos de aplicación al tratamiento de lenguajes, reconocimiento de formas, etc.

Finalmente se vera el  
***Paradigma Evolutivo,***  
en el que *se plantea que*

*la evolución, el crecimiento, la vida, el aprendizaje, el pensamiento, los procesos de descomposición, el desarrollo y transformación de las empresas, sociedades, organizaciones, países, galaxias, universos, etc.,*

*son manifestaciones de un mismo proceso general de transformación o cambio,*

*al que por facilidad llamamos evolución.*

# INTRODUCCIÓN

Los *Sistemas Evolutivos* surgieron como una respuesta a la necesidad de desarrollar *sistemas de información*

(Por ejemplo: nóminas, sistemas expertos, compiladores o sistemas de reconocimiento de imágenes)

*que reflejaran lo más fielmente posible la realidad que están modelando y capaces de soportar y absorber en tiempo real los cambios que ocurren en ésta,*

ya sea en sus elementos, en las relaciones entre éstos o en su significado

Un sistema evolutivo se comporta como un niño que está aprendiendo y aplicando este aprendizaje a su entorno, ya que de entrada,

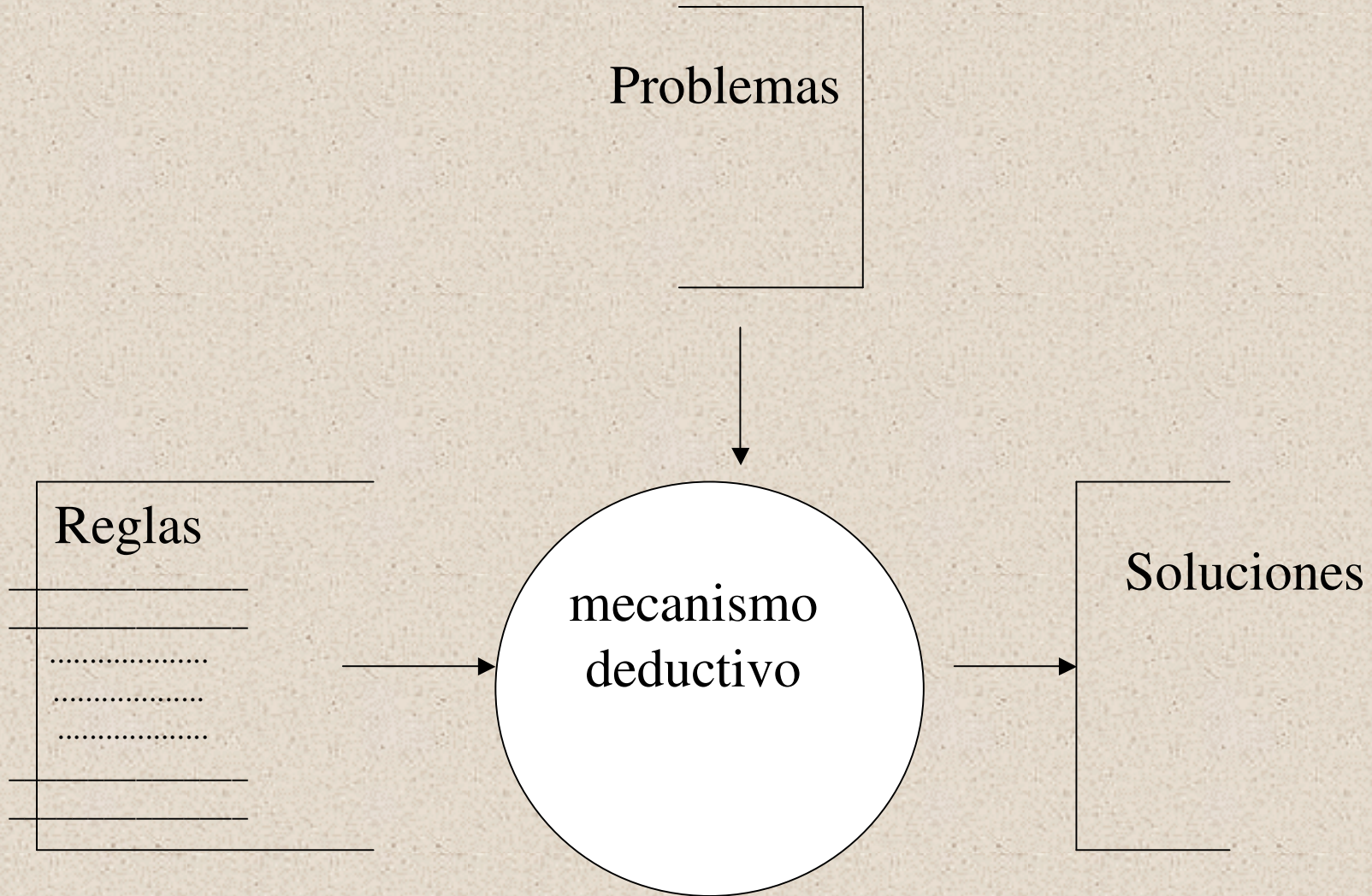
*el Sistema Evolutivo no cuenta con reglas o programas que le digan cómo resolver un problema dado,*

*sino que cuenta con la capacidad de construir su propia imagen de la realidad y con los mecanismos que le permiten percibir esa realidad y actuar dentro de ella.*

**De los Sistemas Formales a los  
Sistemas Evolutivos pasando  
por el Teorema de Gödel**

# Sistemas Formales

años 50



Uno de los problemas más graves de la Informática *poca capacidad para modelar en tiempo real los fenómenos que ocurren en la realidad,*

ya que es común que cuando un Sistema de Información: Nómina, Compilador, Reconocedor de Imágenes, Sistema de Inventarios, Sistema Experto de Diagnóstico Médico, etc. se libera, ya prácticamente es obsoleto,

*porque:*

*1) El problema modelado se modificó.*

*2) Porque el modelo no cubrió los aspectos esenciales.*

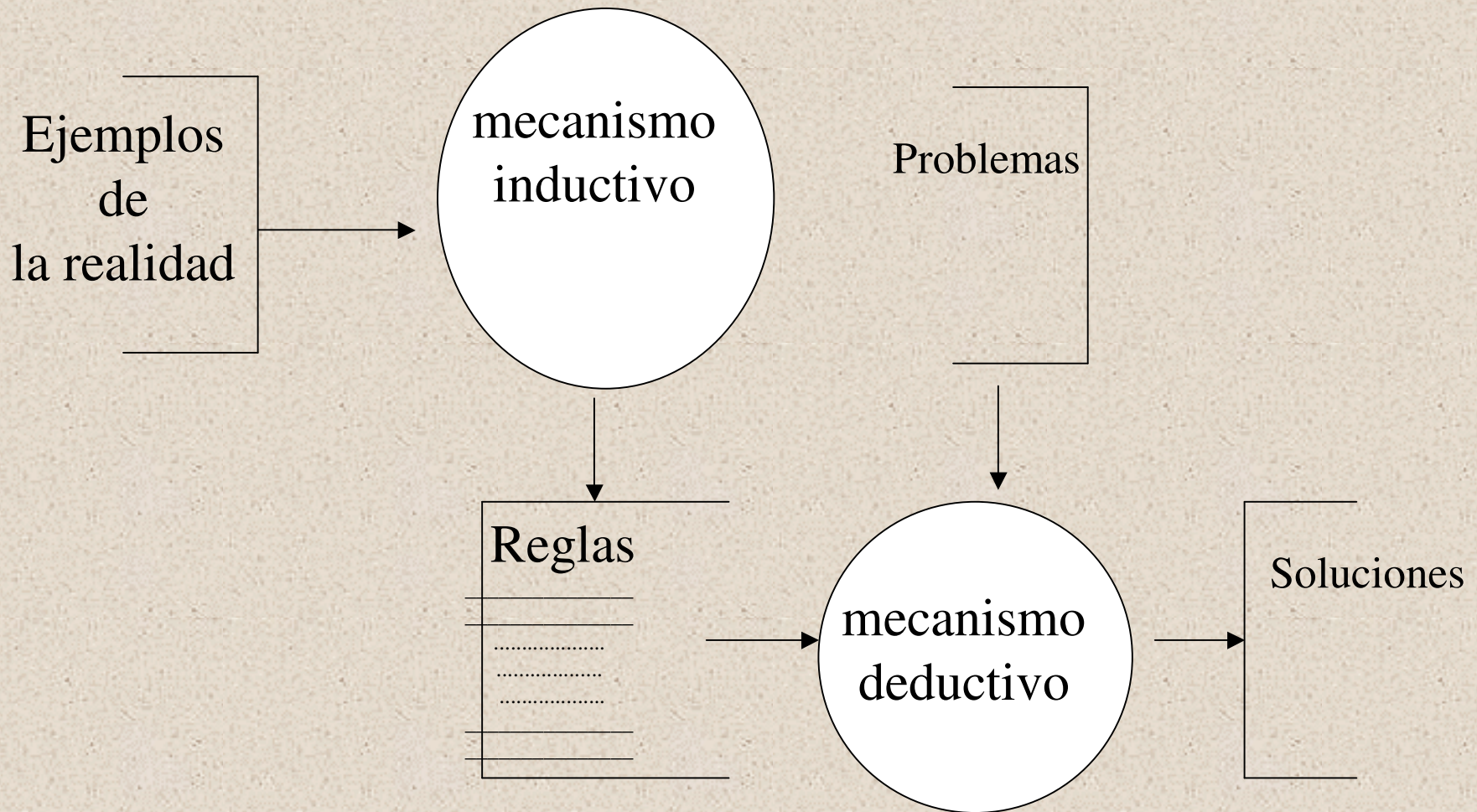
*3) O simplemente la información y el conocimiento que se tiene sobre el tema ha quedado rebasado por algún nuevo dato o hecho no conocido previamente.*

El problema de la programación deductiva o de sistemas formales es que, se orienta a desarrollar sistemas fijos difíciles de modificar en tiempo real (ya que las modificaciones involucran reprogramar el sistema)

Es por lo anterior que es necesario replantear el enfoque utilizado para resolver problemas en Informática (representado por áreas como el Desarrollo de Sistemas, la Ingeniería de Software y la Ingeniería de Conocimiento) en el cual la tendencia es a la construcción de sistemas estáticos e incapaces de automantenerse y buscar métodos y herramientas capaces de recrear en forma continua su imagen de la realidad o del problema a resolver.

desde principios de los 60, se ha buscado desarrollar herramientas capaces de obtener en forma automática el conjunto de reglas del sistema a partir de ejemplos y descripciones generales de un programa, en lo que se conoce como *Machine learning o maquinas o sistemas que aprenden.*

# Machine learning (Sistemas que aprenden) años 60



Las Machine learning tiene dos grandes etapas:  
una “de aprendizaje” y otra Formal.

Por lo que solo resuelve problemas que  
previamente se le enseñó a resolver, pero  
cuando llegan problemas que previamente no  
aprendió no los reconoce,

si por ejemplo se le dan un montón de ejemplos  
de A's reconoce A's, pero si posteriormente se  
me ocurre colocar una A Itálica, ya no la  
reconoce si previamente no se le enseñó a  
reconocer ese tipo de A's.

El problema tanto de los sistemas formales como de los sistemas que aprenden o machine learning es que tarde o temprano les llegara un problema que no pueden resolver.

## **Teorema de Gödel**

En 1930 Kurt Gödel estableció un teorema que lleva su nombre, el teorema de Gödel puede considerarse como un teorema fundamental en informática.

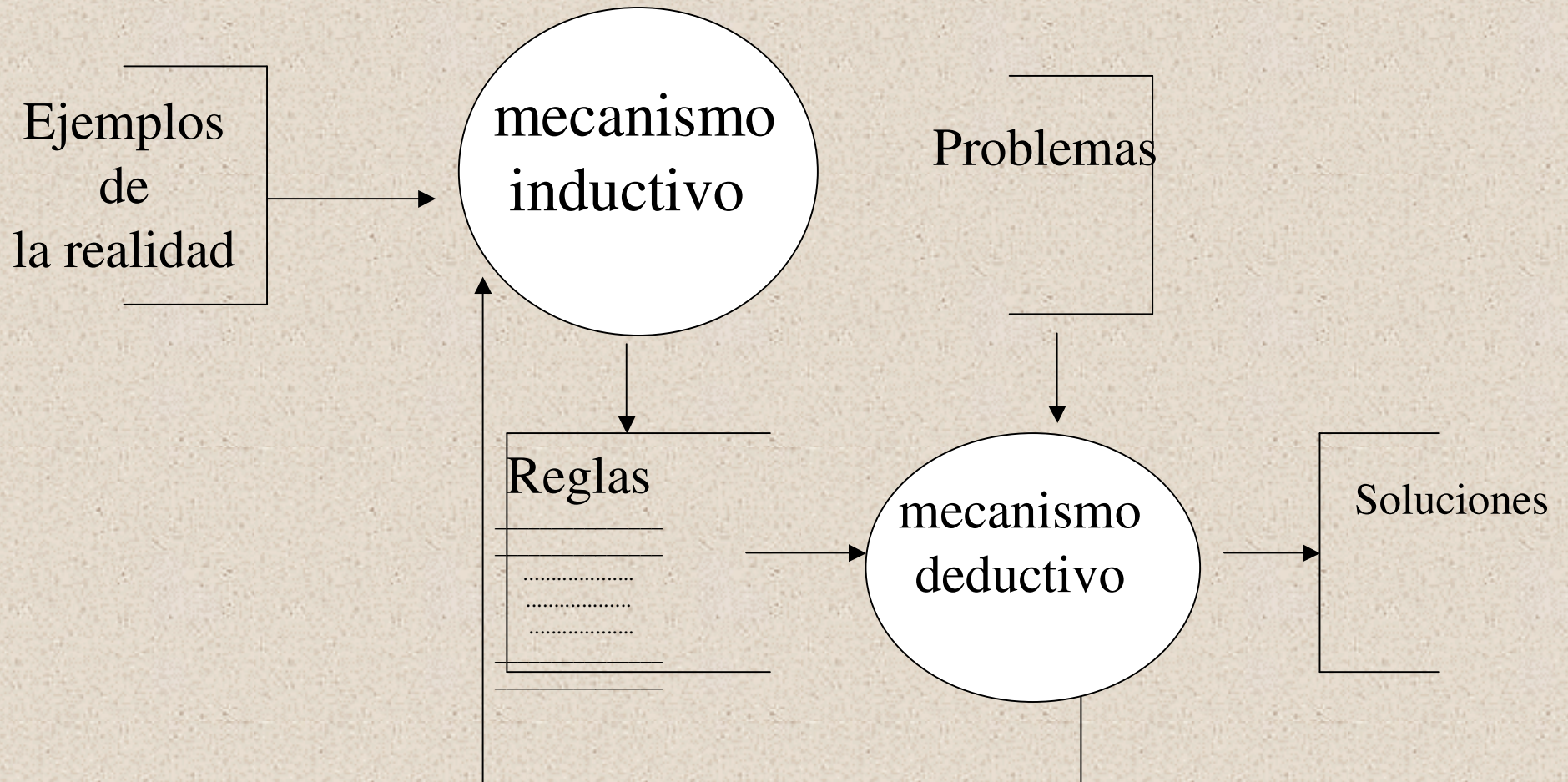
El teorema dice en general que: un sistema Formal, es incompleto, o inconsistente,

Si se logra que un sistema formal sea completo será inconsistente y si es consistente entonces será incompleto.

la idea sería utilizar un programa que por ejemplo contuviera las reglas para reconocer una A, y que cuando no reconociera un tipo especial de A, mandara llamar a otro tipo de programa que se encargara de aprender,

de donde el programa no mandaría Fatal error o warnings, simple y sencillamente si el programa no sabe, aprende.

ese es el principio de lo que es un *sistema evolutivo* o sea un programa que permanentemente esta aprendiendo y reestructurando sus reglas



*Pregunta cuando no  
conoce*

**Sistema Evolutivo básico 1983**

. Retroalimentación para actualizar la imagen de la realidad

mediante los sistemas evolutivos se busca que  
*sea el propio sistema*  
el que lleve a cabo acciones que le permitan construir  
su imagen de la realidad, mantenerla actualizada y  
usarla para interactuar con el medio.

*cuando se desarrolla un  
Sistema Evolutivo  
más que darle un conjunto de reglas prefijadas para  
resolver un problema,  
lo que se busca es  
darle la capacidad para que pueda construir y  
mantener su propia imagen de la realidad.*

# ENFOQUE EVOLUTIVO

un problema se puede resolver mediante los enfoque de los  
*sistemas formales,*  
*sistemas que aprenden*  
*o sistemas evolutivos,*

*Por ejemplo se puede hacer un sistema que reconozca la letra  
A usando el enfoque de los sistemas formales, donde se le da  
al sistema un conjunto de reglas que representan  
completamente a esa A especifica y el sistema reconoce la  
letra A*

también se puede usar el enfoque de los sistemas que aprenden  
y hacer un sistema que aprenda a reconocer letras (por ejemplo  
usando redes neuronales, algoritmos genéticos, etc.)

*o usar el enfoque evolutivo.*

Es un enfoque no una herramienta, por ejemplo existen redes neuronales usando el enfoque formal, el enfoque que aprende o el enfoque evolutivo.

Muchas redes neuronales y algoritmos genéticos son sistemas que aprenden

El enfoque evolutivo se puede aplicar prácticamente con cualquier herramienta incluyendo redes neuronales, algoritmos genéticos y sistemas lingüísticos

**En general los sistemas reales no son formales  
son evolutivos,**

**por lo que la idea es que los sistemas de  
informáticos también sean evolutivos**

# **ejemplos de construcción de sistemas evolutivos**

usando principalmente

herramientas lingüísticas

y

matrices evolutivas

# **SISTEMAS EVOLUTIVOS DE REESCRITURA**

Regla de Reescritura  
 $A \rightarrow B$

Es relativamente fácil construir un sistema de información que representa a los sistemas de reescritura, por ejemplo el siguiente sistema formado por un programa y un archivo con dos columnas :

A1	B1
A2	B2
""	""
An	Bn

Programa()

{i=1

lee **Ax**

mientras ((**Ax != Ai**) y (**no fin de archivo**))i++

si (**no fin de archivo**) escribe **Bi**

sino escribe "no reconozco **Ax**"

}

# Sistemas Evolutivo de Reescritura

A1	B1
A2	B2
""	""
An	Bn

```
Programa()  
{  
  i=1  
  lee Textox  
  mientras ((Textox != Textoi) y (no fin de archivo))i++  
  si (no fin de archivo) escribe Traduccióni  
  sino //entra al dialogo  
    escribe "desconozco Textox dame su traducción"  
    lee Traducciónx  
    almacena en el archivo Textox , Traducciónx  
}
```

# sistema elemental traductor de Idiomas

le damos la entrada "Hola Perro",

si la computadora no encuentra en su lista esta frase, entonces pregunta, ¿ que es "hola perro "?,

le decimos "Hello Dog",

entonces la computadora almacena en su primera columna "Hola perro" y en la otra "Hello Dog",

la siguiente vez que introduzcamos "Hola Perro" la computadora va a decir "Hello dog".

## Traductor automático

tabla que contiene en la primera columna el texto en un idioma y en la segunda su traducción a otro idioma.

<b>texto</b>	<b>traducción</b>
este es un perro	this is a dog
el perro blanco	the white dog
""	""
el perro negro	the black dog

```
Programa()  
{  
  i=1  
  lee Textox  
  mientras ((Textox != Textoi) y (no fin de archivo))i++  
  si (no fin de archivo) escribe Traduccióni  
  sino //entra al dialogo  
  escribe "desconozco Textox dame su traducción"  
  lee Traducciónx  
  almacena en el archivo Textox , Traducciónx  
}
```

Este programa es muy simple de hacer y tiene la ventaja de que no requiere tener almacenado el conocimiento previamente,

si el archivo estuviera vacío y se preguntara por **Textox**

el programa pediría **Traducciónx** y ya tendría la regla **Textox -> Traducciónx**

*el sistema evolutivo puede empezar ha construir la base de conocimiento y 'aprender' desde cero, en tiempo real y fácilmente.*

Una gran cantidad de problemas aparentemente diferentes se pueden reducir en primera instancia al mismo sistema, ya que prácticamente se puede atacar cualquier problema que se pueda representar como una cadena de bits o caracteres, como por ejemplo *el reconocimiento de imágenes, texto, traductores, sistemas experto, señales biofísica, voz, etc.*

# **INDEPENDENCIA RELATIVA**

En general se considera que un Sistema de Información o programa de cómputo tiene la arquitectura de la figura



Arquitectura de los sistemas de información en los 50s

ya para finales de los 70's se manejaba un modelo generalizado en el cual se considera que cualquier sistema de información tiene la arquitectura de la siguiente figura



Arquitectura de los sistemas de información en los 70s

El desconocimiento o el hecho de no tomar en cuenta este esquema cuando se desarrollan sistemas es una de las causas principales por las que los sistemas se vuelven altamente estáticos y difíciles de mantener

ya que en los sistemas y programas tradicionales las tres componentes se encuentran revueltas, por lo que, un cambio "pequeño" en los datos o procesos o en el orden de atacar un problema ocasiona que prácticamente se tenga que volver a programar todo;

por otro lado si se desarrolla el sistema de tal manera que los datos queden en un lado, los procesos en otro y finalmente la estructura del sistema en otro, el proceso de actualización puede ser relativamente fácil (un caso particular de este enfoque es el del desarrollo de Bases de Datos).

una característica fundamental que se debe buscar cuando se desarrolla un sistema de información es la de que exista una

## *Independencia Relativa*

*los datos, procesos y estructura del sistema queden separados y únicamente existe la relación mínima necesaria entre las tres componentes.*

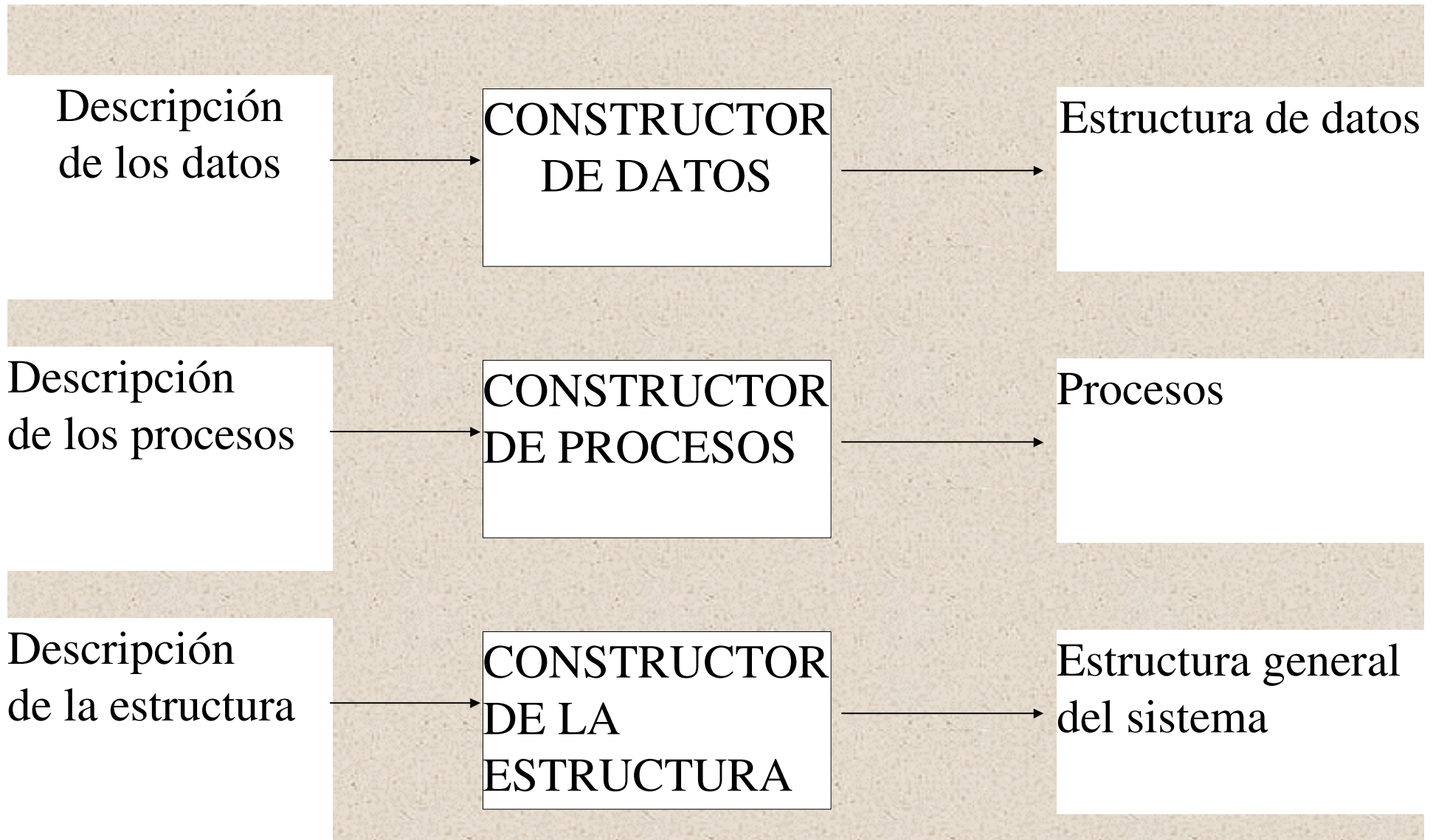


Figura 4. Constructores de datos, procesos y estructura

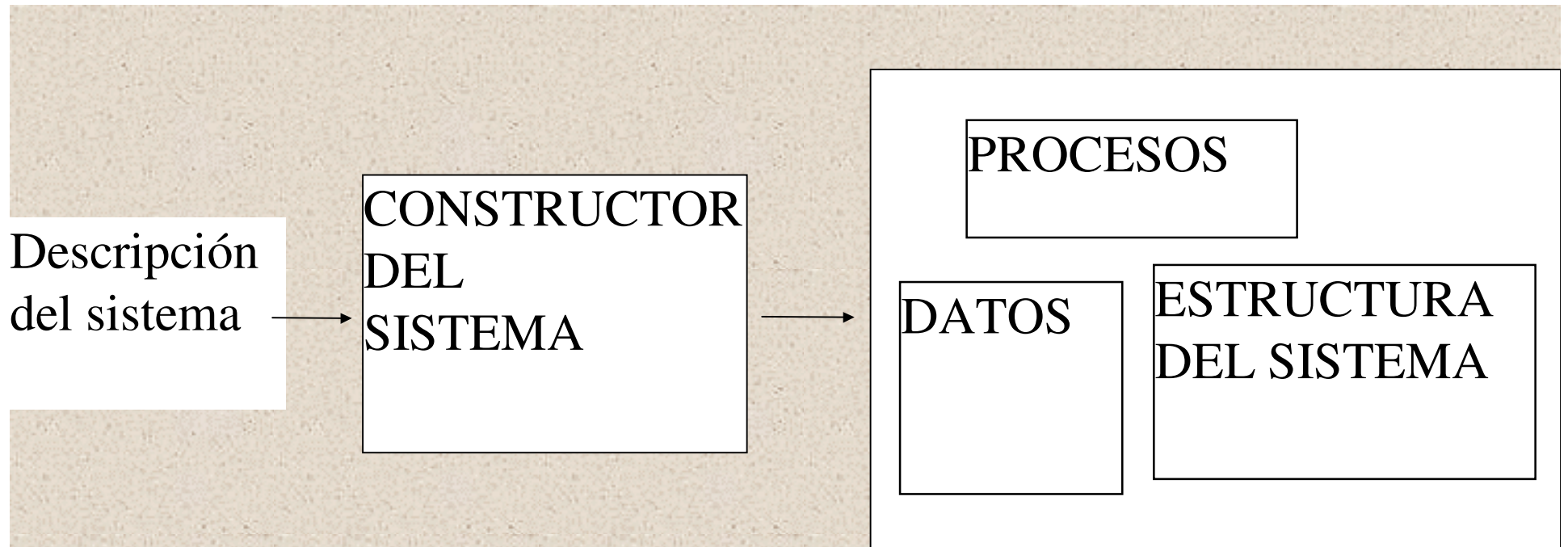


Figura 5. Integración de los constructores en un constructor del sistema

El constructor del sistema es un programa capaz de construir la estructura general de un Sistema de Información a partir de la descripción del sistema, de la misma forma que un constructor de Base de Datos es capaz de construir el esquema físico de una Base de Datos a partir de un Esquema Lógico de Base de Datos.

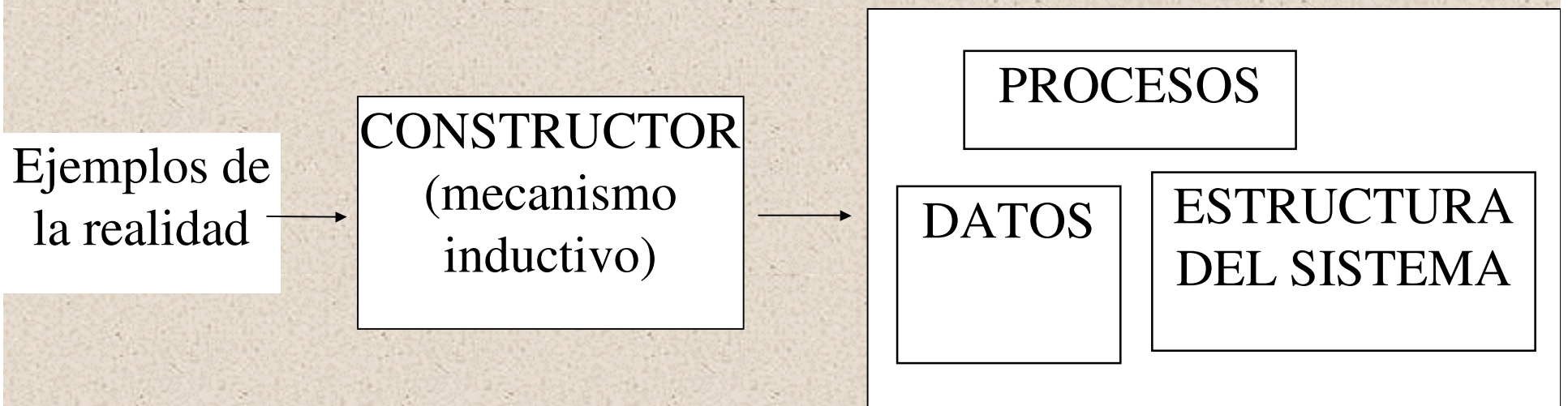
Si se construye el sistema de información de tal forma que los componentes sean independientes en forma relativa entre sí

y se da una interrelación entre el constructor y el sistema de tal forma que cualquier cambio en la descripción del sistema se refleje en tiempo real en el sistema de información,

entonces se puede considerar que la imagen de la realidad que tiene el sistema de información es bastante cercana a la que se quiere reflejar.

A una herramienta o programa capaz de encontrar un conjunto de reglas a partir de ejemplos la conocemos como Mecanismo o Herramienta Inductiva;

el esquema de un constructor basado en una herramienta inductiva es el mostrado en la figura



Herramienta inductiva

En el esquema anterior se maneja un mecanismo inductivo para construir una Imagen de la Realidad, sin embargo en esta imagen solo se tienen un conjunto de reglas en términos de procesos, estructura y datos del sistema por lo que es necesario integrar una herramienta capaz de resolver problemas siguiendo esas reglas, o sea un mecanismo de tipo deductivo como se ve en el diagrama

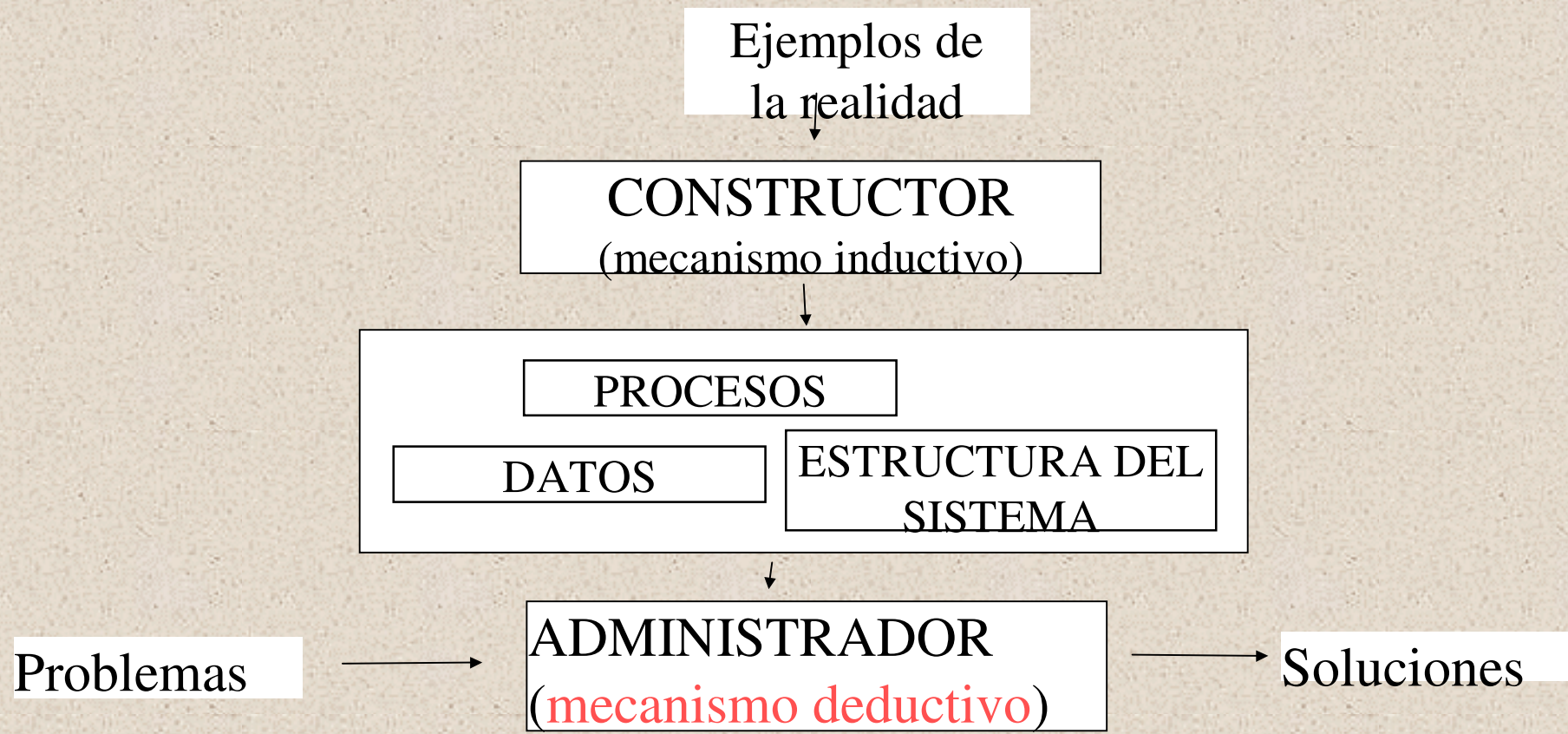
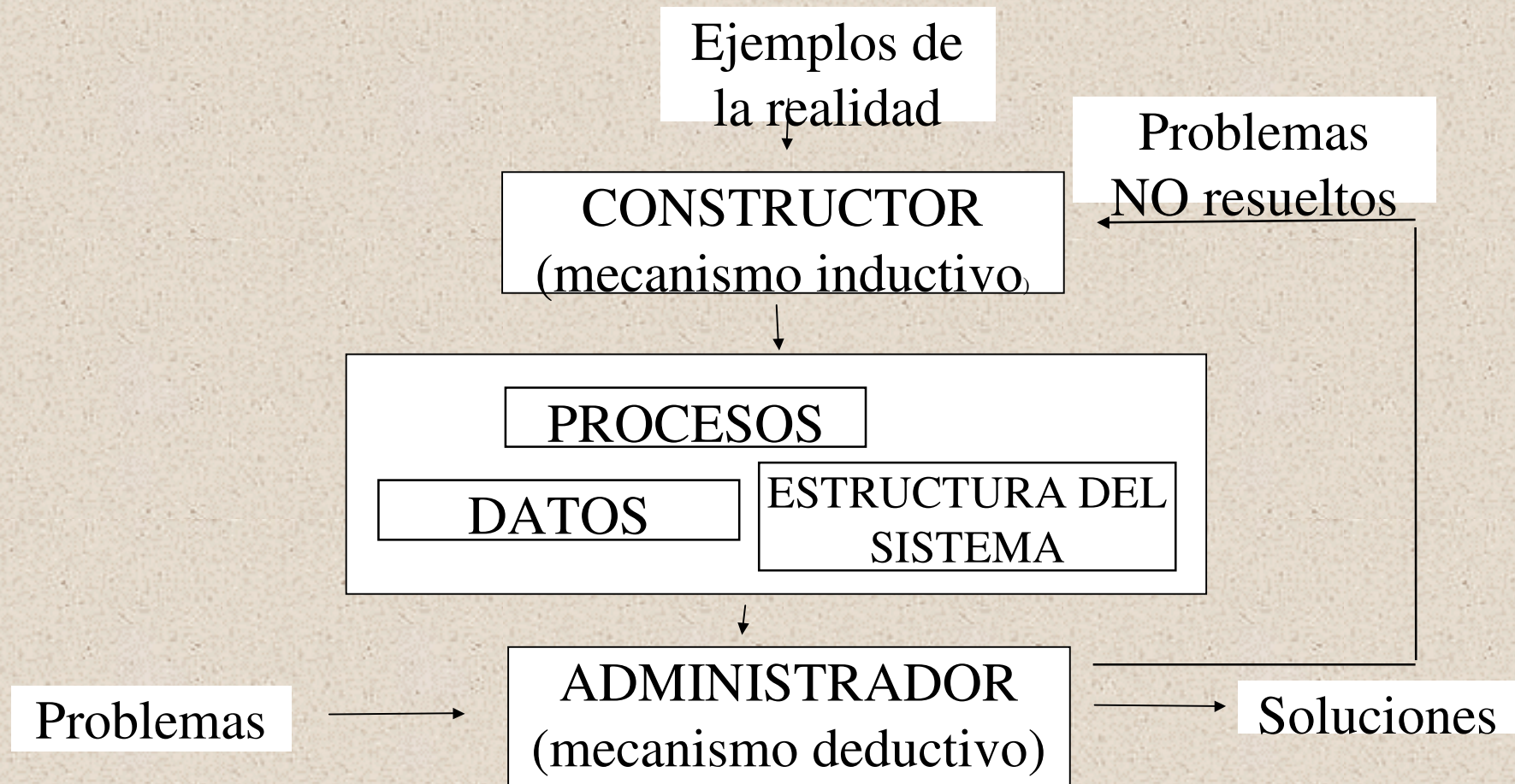


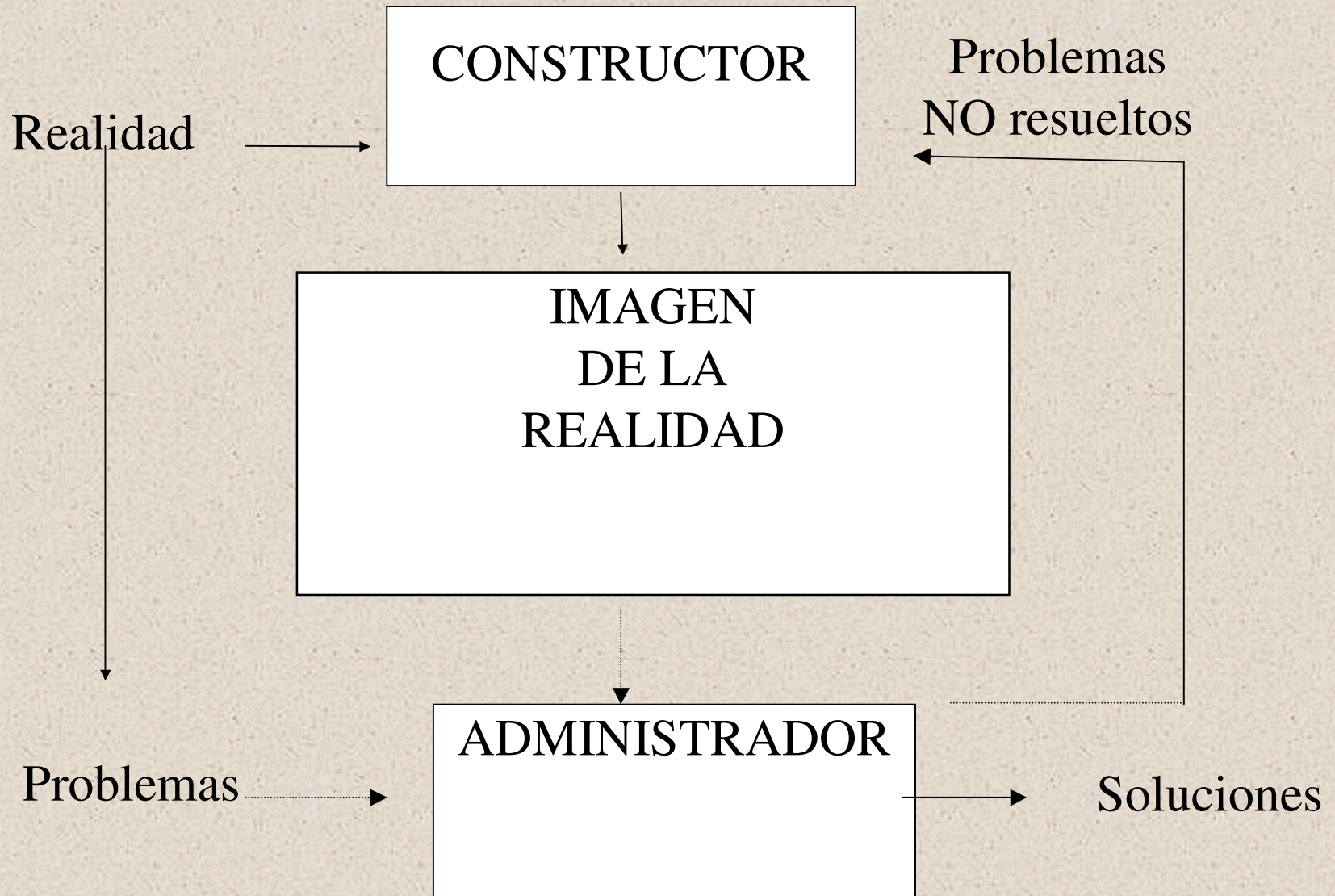
Figura 7. Integración de la componente deductiva

Para contar con un **Sistema Evolutivo** lo único que hace falta es dotar a esta herramienta con la capacidad de actualizar su imagen de la realidad en tiempo real con lo que se tiene la arquitectura de la figura.



Retroalimentación para actualizar la imagen de la realidad

# Arquitectura General de un Sistema Evolutivo



# Enfoque Lingüístico

una forma de ver la realidad,  
en la cual se considera que,

*cualquier cosa se puede ver como una oración de algún  
lenguaje.*

## *Enfoque Lingüístico*

Mediante este enfoque se pueden representar como oraciones de algún lenguaje las imágenes, las reglas de un sistema experto, las trayectorias de un planeta, el movimiento de la mano, los sentimientos, la trayectoria que sigue una pieza de ajedrez al moverse, una huella digital, la señal de un electrocardiograma, etc.,

con lo que se amplía el concepto de lenguaje que normalmente se restringía a los lenguajes naturales (como el Español, Inglés, Chino o Árabe) y artificiales (como Fortran, Pascal o C), para incluir cualquier cosa.

*se partirá de que*

*el lenguaje de comunicación con el exterior  
(o sea el lenguaje con el que se plantean los  
problemas al Sistema Evolutivo y con el cual  
éste responde a los requerimientos)*

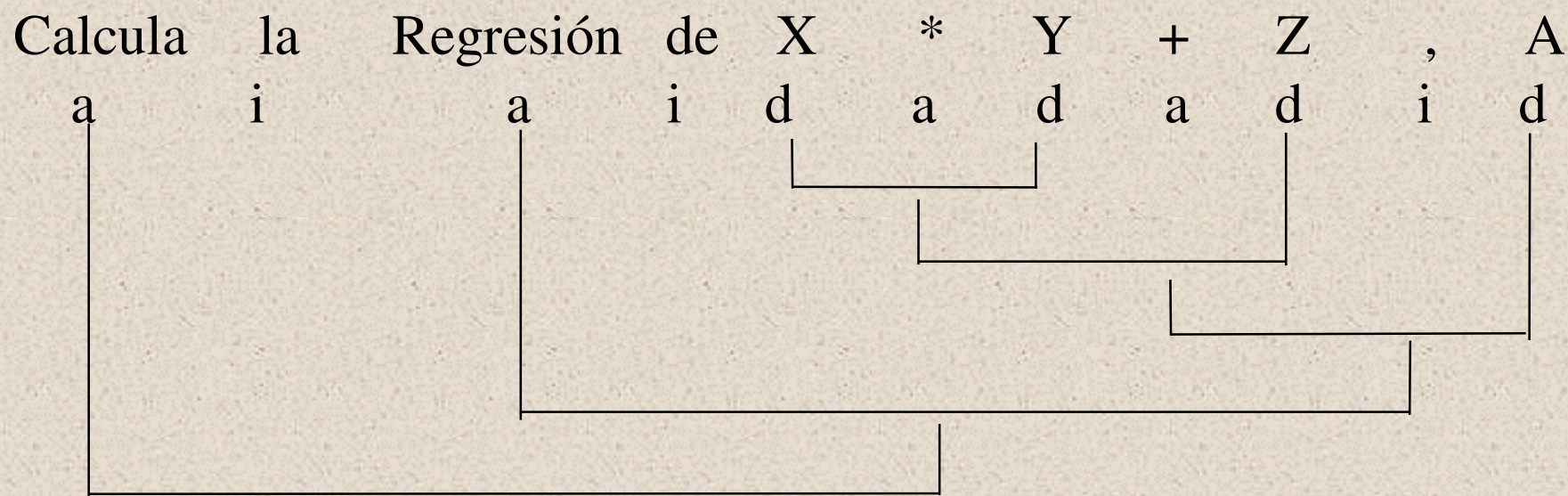
*y el lenguaje con el que el Sistema Evolutivo  
construye su imagen de la realidad*

*son el mismo.*

por ejemplo en la oración:

Calcula la Regresión de  $X * Y + Z$ , A

Es relativamente fácil detectar los datos, acciones y estructura:



la estructura se indica con las líneas y representa el orden en que se ejecutan las acciones sobre los datos, los datos tienen el tipo **d**, las acciones el tipo **a** y las palabras no relevantes llevan **i** (ignora).

un Sistema Evolutivo debe tener la capacidad de encontrar los componentes del sistema de información a partir del lenguaje utilizado en el área problema

Un sistema lingüístico tiene involucrados al menos

Datos

elementos o componentes léxicos (unidades léxicas)

estructura

sintaxis

Procesos

componentes semánticos (significado),

por lo que a continuación veremos en general como encontrar e integrar esos componentes a partir del, lenguaje utilizado en el área problema

## Manejador Léxico

El Manejador Léxico es un programa que recibe como entrada una oración en algún lenguaje y obtiene las unidades léxicas presentes en la oración, el tipo de cada unidad léxica y la estructura general u oración canónica.

Por ejemplo, en la oración

Calcula la Regresión de X

se encuentran las Unidades Léxicas mostradas en la tabla.

Calcula	A1 , acción
la	i, ignora
Regresión	a2
de	i
X	d1, dato

Calcula la Regresión de X  
a1 i a2 i d 1

A

a1 i a2 i d 1

se les conoce como oración canónica

en general cuando se sustituyen los elementos de una oración por una representación que permita visualizar la estructura de la oración se obtiene una oración canónica.

Originalmente el Manejador Léxico no tiene ningún conocimiento acerca del lenguaje a utilizar y solamente cuenta con la capacidad para encontrar los diferentes tipos de unidades léxicas de un lenguaje.

cotidianamente manejamos una palabra como una cadena de caracteres (letras, números, signos) separados por espacios, por ejemplo en la oración “el perro ladra” tenemos tres palabras: “el”, “perro”, “ladra”, el problema es que estamos manejando elementos semánticos y los elementos semánticos no necesariamente son cadenas separadas por blanco, por ejemplo “Buenos Aires” es un elemento semántico que se refiere a una ciudad, “buenos” es otro elemento y “aires” es otro, en el manejador es conveniente tratar de manejar elementos semánticos y no solo palabras, a la unidad con significado semántico propio se le llama unidad léxica la unidad léxica corresponde en muchos casos a una palabra, pero existen otros muchos casos en los cuales eso no es cierto

Las dos variantes mas generales del método son:

Tomar un texto y buscar la cadena mas grande que se repita en el texto y sustituirla por una etiqueta X1,  
tomar la siguiente cadena que se repita y sustituirla por una cadena X2

y así sucesivamente mientras existan cadenas,  
con este método asumimos que cada  $X_i$  es una “unidad léxica”

Otro método parte al revez,  
se buscan las palabra (cadenas separadas por blancos) y a todas las palabras iguales se les asigna la misma etiqueta,  
luego si se encuentra una combinación de etiquetas repetida varias veces

por ejemplo la cadena de etiquetas X8X3X14 aparece repetida varias veces, se supone que toda esa combinación también es por si sola una unidad léxica y se le asigna su propia etiqueta.

# Manejador Sintactico

A partir de los resultados del Manejador Léxico el Manejador Sintáctico encuentra la estructura del sistema para lo cual, toma como entrada el conjunto de oraciones canónicas y genera la Estructura.

Este mecanismo es tal vez el componente mas importante de un Sistema Evolutivo basado en lenguajes, ya que, es el responsable de encontrar las reglas generales o patrones de estructura del sistema y para lograrlo utiliza normalmente métodos de la inferencia gramatical.

# Inferencia Gramatical

El problema que ataca la Inferencia Gramatical consistir básicamente en encontrar la Gramática (Estructura) que describe a un lenguaje dado a partir de ejemplos de oraciones del Lenguaje.



Inferencia gramatical

Las herramientas de la Inferencia Gramatical trabajan  
con la estructura de las oraciones

buscando encontrar una estructura general  
(o reglas sintácticas)

a partir de estructuras particulares  
(u oraciones).

Muchos de los métodos de inferencia gramatical se basan en las operaciones básicas de  
Factorización,  
Distribución y  
Recursividad

por lo que a continuación se explicaran estas operaciones.

# Factorización Lingüística

La factorización lingüística consiste en la búsqueda de cadenas que se repiten varias veces en un texto y verlas como un factor dentro del texto.

Así, si por ejemplo se tienen las siguientes oraciones:

Juan es hermano de Pedro y  
Juan estudia en UPIICSA

se puede detectar que en las dos oraciones se encuentra presente la palabra Juan y que un párrafo equivalente sería:

Juan es hermano de Pedro y estudia en UPIICSA

Si se observa lo que se ha hecho es detectar que la palabra Juan era común a las dos oraciones por lo que se factorizó (o sea que se sacó como factor común) y se obtuvo un párrafo donde sólo aparece una sola vez.

Para que se pueda visualizar el proceso sustituiremos fragmentos de la oración por etiquetas de acuerdo a la siguiente tabla:

Fragmento	Etiqueta
Juan	o1
es hermano de	r1
Pedro	o2
y	+
estudia en	r2
UPIICSA	o3

Con lo que el párrafo

<u>Juan</u>	<u>es hermano de</u>	<u>Pedro</u>	<u>y</u>	<u>Juan</u>	<u>estudia en</u>	<u>UPIICSA</u>
o1	r1	o2	+	o1	r2	o3

quedaría como:

o1 r1 o2 + o1 r2 o3

Donde se observa que **o1** es común a las 2 oraciones.

Recordemos que la factorización algebraica consiste en encontrar los factores comunes en una expresión algebraica y sacarlos de la expresión, como se ve a continuación:

$$\underline{a}b + \underline{a}c = a(b+c)$$

$$3x + 3y = 3(x + y)$$

$$.a(b * c) + a(e/f) = a(b*c+e/f)$$

Aplicando lo anterior a las oraciones canónicas  
entonces tenemos que:

$$\underline{o1} \ r1o2 + \underline{o1} \ r2o3 = \underline{o1}(r1o2+r2o3)$$

o sea que el párrafo

$$o1r1o2 + o1r2o3$$

es equivalente al párrafo

$$o1 (r1o2+r2o3)$$

si sustituimos las etiquetas por los fragmentos que  
representan tenemos entonces que:

$$\begin{array}{ccccccc} \underline{\text{Juan}} & \underline{\text{es hermano de}} & \underline{\text{Pedro}} & \underline{\text{y}} & \underline{\text{estudia en}} & \underline{\text{UPIICSA}} & \\ o1 & r1 & o2 & + & r2 & o3 & \end{array}$$

Una aplicación que se desarrolló en 1988 en colaboración con Javier Ortiz (en esa época Coordinador de la Maestría en Computación del CENIDET en Cuernavaca, Mor.)

consistió en la aplicación de la factorización a la

construcción de un

Sistema Evolutivo generador de Sistemas Expertos,

en el cual, la idea consistió básicamente en

tomar una gran cantidad de oraciones en las cuales un experto

explica como resuelve un problema y

transformar cada oración en una oración canónica incluyendo

por ejemplo

síntomas (s), diagnósticos (d) y tratamientos (t)

e ignorando todo lo demás.

Por ejemplo si se tiene la oración:

Paciente femenino de 15 años con 38 grados de temperatura y  
S1 S2 S3  
dolor en el pecho, se le diagnosticó faringitis y se le  
S4 d1  
recetó antibióticos, antistamínicos y reposo  
t1 t2 t3

la oración canónica equivalente sería:

S1 S2 S3 S4 d1 d1 t1 t2 t3

si en lugar de tener una sola oración  
se tiene la información de todos los  
pacientes del hospital

se pueden obtener cientos o miles de reglas  
canónicas,

las cuales mediante factorización pueden  
proporcionar las reglas generales de un  
problema y su tratamiento.

Por ejemplo si se tienen las reglas.

$S \rightarrow \underline{S1} S2S3S4d1t1t2t3$   
 $\quad \underline{S1} S2S5d2t1t4$   
 $\quad \underline{S1} S4S6S7d3t5$

Factorizando S1

$S \rightarrow S1X$   
 $X \rightarrow S2 S3S4D1T1T2T3$   
 $\quad S2 S5D2T1T4$   
 $\quad S4S6S7D3T5$

Factorizando S2

$S \rightarrow S1X$   
 $X \rightarrow S2Y$   
 $\quad S4S6S7D3T5$   
 $Y \rightarrow S3S4D1T1T2T3$   
 $\quad S5D2T1T4$

## Distribución Lingüística

La distribución lingüística es la operación inversa de la factorización y para entenderla veremos un ejemplo basado en problemas de representación del conocimiento.

Existen varios mecanismos para representar el conocimiento, pero uno de los más usados son las redes semánticas, donde una red semántica está formada por hechos o reglas de la forma **objeto relación objeto (o r o)**, por ejemplo a partir de la oración: **Juan es hermano de Pedro**

se detecta que **Juan** es un *objeto*, **es hermano de** es una *relación* y **Pedro** es otro *objeto* por lo que de esta oración se obtienen directamente los componentes de una red semántica

Jesús Olivares en su tesis de licenciatura  
"Sistemas Evolutivos para Representación de Conocimiento"  
desarrollada en la UPIICSA del IPN,  
planteo un mecanismo evolutivo para construir un base de  
conocimientos basada en una Red Semántica Aumentada  
aplicar una serie de técnicas de Inferencia Gramatical  
basadas en la distribución lingüística,  
para transformar una oración compuesta de múltiples  
elementos  
en un conjunto de oraciones semánticamente equivalentes a la  
original,  
donde cada oración final es de la forma **oro**,  
por lo que, son fácilmente representadas con una red  
semántica.

## Por ejemplo en la oración:

Juan estudia en el Poli, trabaja en el INEGI y vive en Lindavista.

Juan	es un objeto o0
estudia en el	es una relación r1
poli	es un objeto o1
,	es un separador +
trabaja en el	es una relación r2
INEGI	es un objeto o2
y	es un separador +
vive en	es una relación r3
Lindavista	es un objeto o3

y queda representada como:

**o0** ( r1o1 + r2o2 + r3o3 )

El paréntesis que esta después de **o0** refleja que tiene relación con todos los demás elementos de la oración.

se tienen varios hechos pero sin embargo no son directamente representables mediante una red semántica, por lo que se le aplicara la distribución lingüística,

Si se toma la oración y se le aplica la operación de distribución del álgebra queda:

$$o(r_1o_1 + r_2o_2 + r_3o_3) = o_1r_1o_1 + o_2r_2o_2 + o_3r_3o_3$$

y si sustituimos el significado de los diferentes elementos tenemos que:

$o_1r_1o_1$  Juan estudia en el poli  
+  
 $o_2r_2o_2$  Juan trabaja en el INEGI  
+  
 $o_3r_3o_3$  Juan vive en Lindavista.

o sea que a partir de una oración compleja se obtuvieron tres oraciones simples del tipo **oro** que mantienen el significado de la oración original y que son directamente representables en una red semántica.

Cuando nos comunicamos factorizamos  
Cuando almacenamos distribuimos

Jesús Olivares Ceja

# **MATRICES EVOLUTIVAS**

A mediados de los 80's, Cuitláhuac Cantú, al estar trabajando sobre reconocimiento de imágenes llegó a una representación matricial con capacidades evolutivas

En esta idea se parte de que originalmente la matriz está vacía y lo que hace el sistema es buscar una imagen, como no encuentra nada la coloca en el primer renglón, cuando llega la segunda imagen, si son similares la reconoce y la acumula con la primera si no son similares entonces la coloca en el siguiente renglón y así sucesivamente.

cada imagen se representa como un vector,

de tal forma que por ejemplo,

si se tiene un gato, un perro y un ratón,

cada uno de ellos se almacena como un vector

y entre los tres forman una matriz como la siguiente

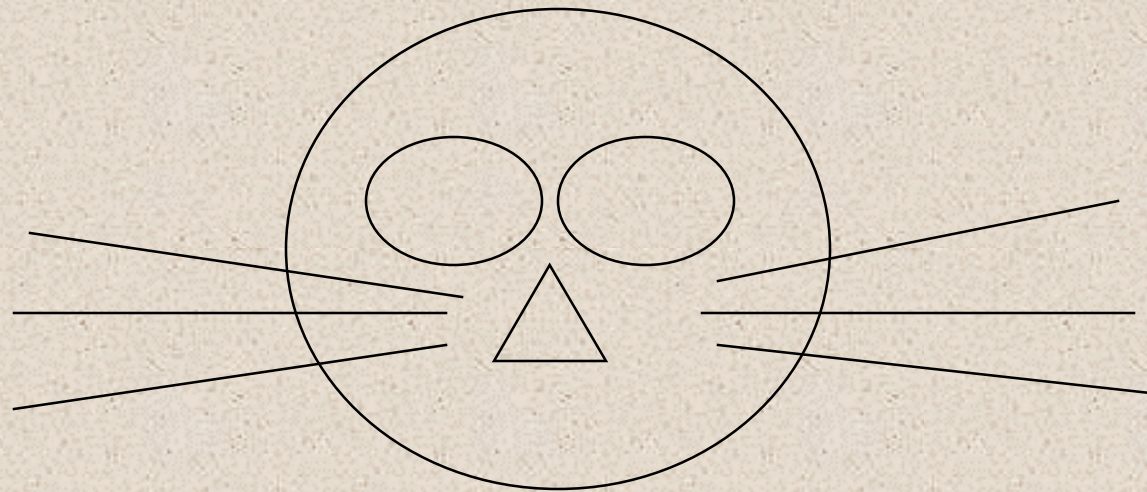
a1	a2	a3	a4	a5	a6	a7	a8	a9	
1	0	1	0	0	1	0	1	0	gato
0	1	0	1	0	1	1	0	1	perro
1	1	0	0	1	0	0	1	1	ratón

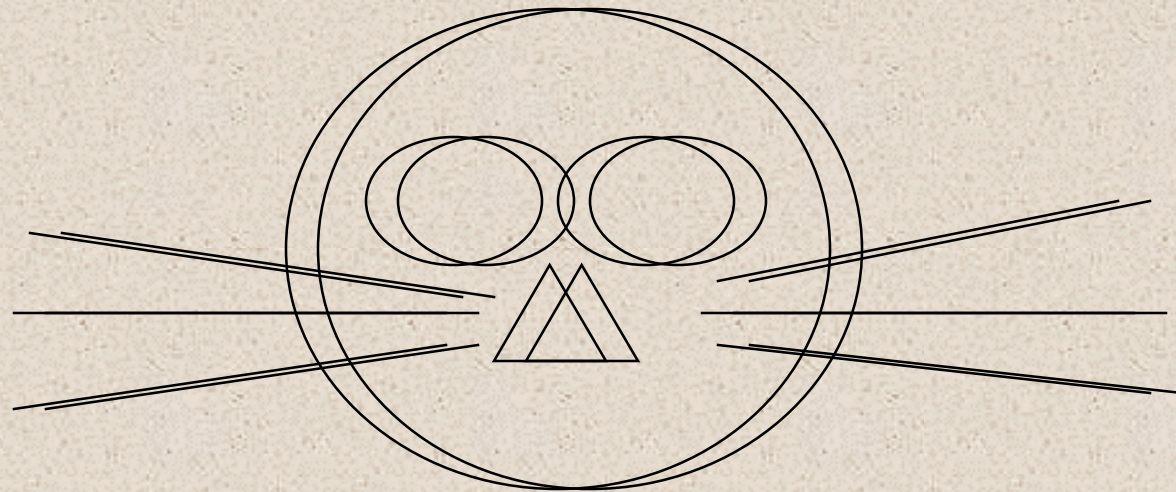
Si se perciben varios gatos,  
en lugar de almacenar cada gato en un vector  
independiente

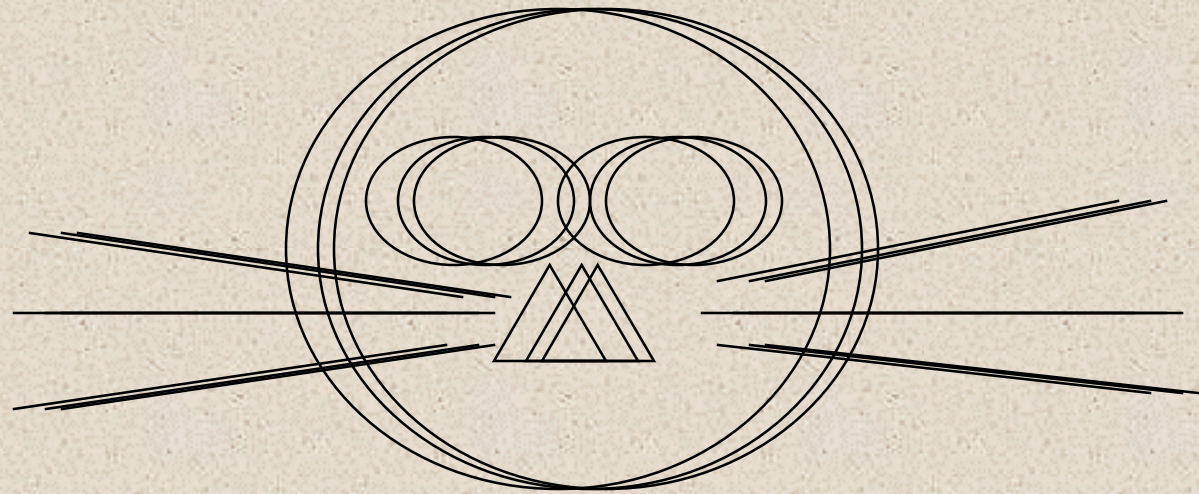
se suman los vectores que representan cada uno de los  
gatos y el resultado se toma como la representación de  
la estructura general del gato  
y se almacena en la matriz.

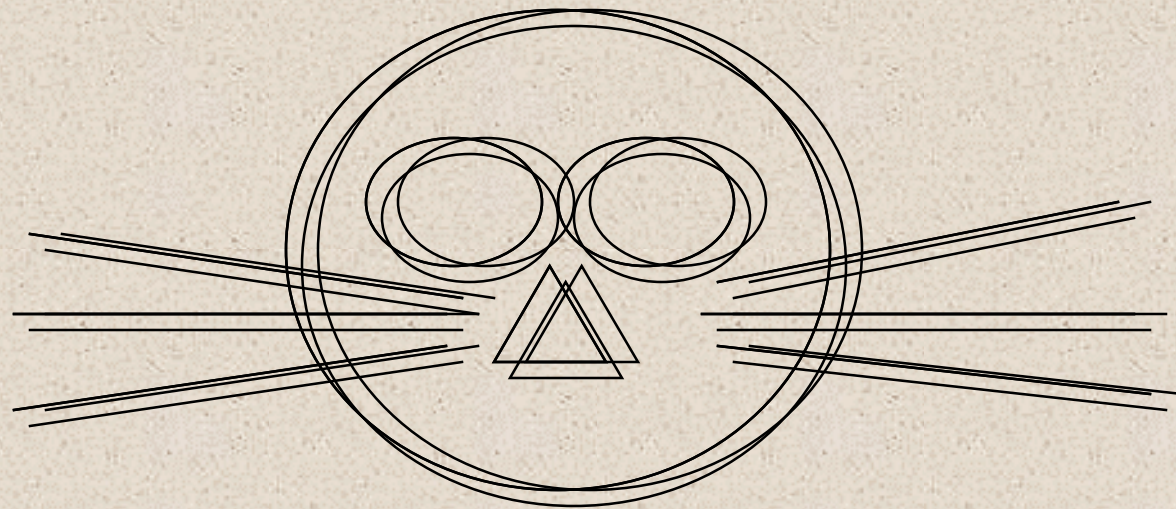
( 5 1 4 0 0 5 0 4 1 20 ) gato

Lo anterior es equivalente a que cada gato se dibujara en un  
acetato y posteriormente se sobrepusieran los acetatos, con lo  
que las características repetitivas del gato quedan más  
recalcadas y equivale a números mayores en el vector que lo  
representa.









Por otro lado si se tienen varios gatos, perros y ratones, la matriz seria de la forma:

a1	a2	a3	a4	a5	a6	a7	a8	a9	h	
5	1	4	0	0	5	0	4	1	20	gato
0	3	0	3	0	2	3	2	1	14	perro
1	1	0	0	1	0	0	1	1	5	ratón

cada renglón representa un tipo de objeto y  $h$  es un valor donde se acumula el número de puntos en el vector

(En este caso se puso  $h$  como la suma de los valores del vector por facilidad del ejemplo, sin embargo existen otros métodos para asignar este valor).

Si se desea reconocer un nuevo objeto, se representa también como un vector, se multiplica por la matriz, se ve en que renglón se obtuvo el máximo valor y se le asocia a ese renglón el objeto.

Por ejemplo si tomamos la matriz anterior y llega el objeto 1 0 1 0 0 1 0 1 0, se multiplica por la matriz y les restamos el valor de  $h$  quedando:

-2 para el gato, -10 para el perro y -3 para el ratón.

De donde se propone que el objeto reconocido es un gato.

*Como siguiente punto el sistema acumula el nuevo vector en la matriz, con lo que, en el ejemplo anterior el nuevo vector se suma al renglón del gato y la matriz queda:*

a1	a2	a3	a4	a5	a6	a7	a8	a9	h	
6	1	5	0	0	6	0	5	1	24	gato
0	3	0	3	0	2	3	2	1	14	perro
1	1	0	0	1	0	0	1	1	5	ratón

*Con lo que, el sistema evolutivo esta transformando permanentemente su imagen de la realidad.*

Como se puede observar este método reúne la característica de que esta evolucionando en forma natural y encontrando la imagen acumulada (y por tal, la imagen promedio),

*con lo que no existe un proceso previo de aprendizaje y otro de aplicación, sino que por el proceso natural de conocer y reconocer las imágenes va evolucionando.*

Un ejemplo donde se pueden aplicar las matrices evolutivas es en el área de los sistemas expertos,

Si partimos de un sistema experto de diagnostico formado por reglas de la forma

$$S_1 \ S_2 \ S_3 \ \dots \ S_n \ \Rightarrow \ D_i : T_i$$

donde  $S_j$  son los síntomas,  $D_i$  los diagnósticos y  $T_i$  los tratamientos (un conjunto de síntomas generan un diagnostico y un tratamiento)

El sistema experto puede representarse como una matriz, cada columna corresponde a un síntoma y cada renglón a una regla del experto.

	S1	S2	S3	...	Sn
r1	1	0	0	...	0
r2	0	1	1	...	0
...	...	...	...	...	...
rm	1	0	1	...	1

De tal manera que si aparece un 1 en el renglón  $i$  y la columna  $j$ , quiere decir que la regla  $i$  necesita el síntoma  $j$  para cumplirse.

Por ejemplo, si se tiene la siguiente matriz:

	S1	S2	S3	S4	S5	S6
regla1	3.2	0	4.0	6.5	0	0
regla2	2.0	1.3	0	0	1.5	0
regla3	0	3.0	0	0	4.2	2.0

y llega un problema con los síntomas

(1.0 0 1.5 0 0 0.7)

entonces se multiplica la matriz por el vector quedando el resultado:

regla1	9.2
regla2	2.0
regla3	1.4

de donde se propone la regla1 como el resultado del sistema.

Una de las ventajas de representar al sistema experto mediante una matriz es que ahora se puede ver y atacar con múltiples herramientas incluyendo entre otras la lógica difusa, redes neuronales, teoría de la medida y sistemas evolutivos.

En particular podemos aplicar las técnicas de matrices evolutivas

partir de que la matriz está originalmente vacía

cuando llegan los primeros síntomas y no encuentra nada almacena el vector en el primer renglón y asigna el diagnostico (le puede pedir el diagnostico al experto humano).

Por ejemplo, si llegan los síntomas

(1.2 0 0 2.3 0 0.7)

el sistema busca en la matriz el diagnostico, pero como esta vacía no los encuentra, entonces pregunta por el diagnostico y genera una matriz con un renglón.

	S1	S2	S3	S4	S5	S6	
r1	1.2	0	0	2.3	0	0.7	D1

Si llegan ahora los síntomas  
(0.7 0 0 0 3.2 0),  
como la colisión con el vector almacenado es  
baja, pregunta por el diagnostico,  
si el nuevo diagnostico es diferente crea un  
nuevo renglón,  
quedando la matriz:

	S1	S2	S3	S4	S5	S6	
r1	1.2	0	0	2.3	0	0.7	D1
r2	0.7	0	0	0	3.2	0	D2

Si llega una cadena de síntomas con alta colisión con D1,  
como por ejemplo:

(1.0 0 0 1.8 0 1.2)

el sistema emite el diagnostico D1 y acumula el nuevo síntoma  
sobre la matriz anterior en el renglón 1:

	S1	S2	S3	S4	S5	S6		# de acumulados
r1	2.2	0	0	4.1	0	1.9	D1	2
r2	0.7	0	0	0	3.2	0	D2	1

Al tener el vector acumulado se están reforzando los síntomas  
y en forma natural cambian los pesos.

El sistema siempre suma.

Si llega una cadena de síntomas con alta colisión los suma al renglón diagnosticado.

Si llega una nueva cadena crea un nuevo renglón.

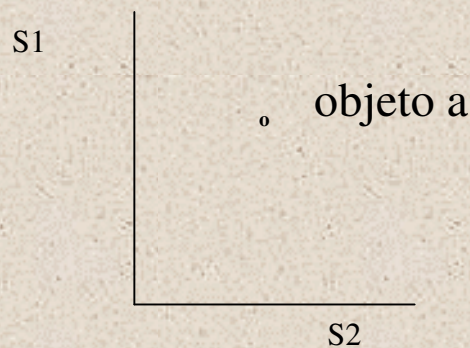
Si llega una cadena y el experto indica que corresponde a un diagnóstico presente, la nueva cadena se acumula a este diagnóstico.

Si se toma el vector acumulado y cada síntoma lo dividimos entre el número de acumulados tenemos una regla promedio, por lo que, lo que estamos encontrando es la regla promedio asociada a un diagnóstico

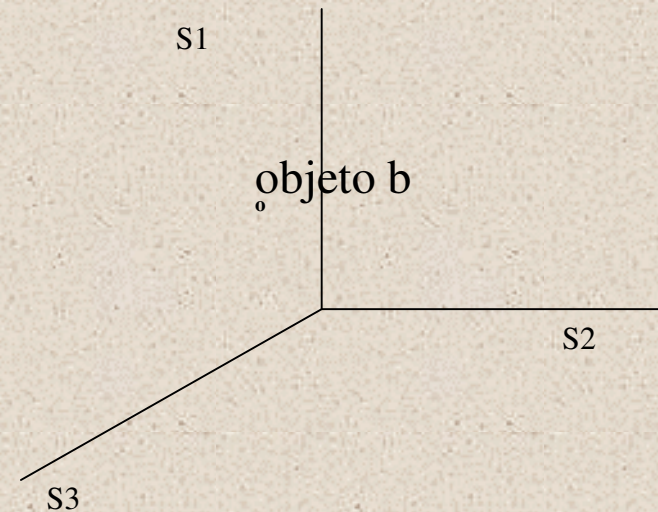
y mediante el mecanismo evolutivo esta regla se está depurando y afinando cotidianamente, o sea que, en forma natural y permanente se están encontrando y afinando las reglas del sistema.

# **Una Representación N-Dimensional**

Si analizamos un poco mas a las matrices evolutivas vemos que cada imagen, regla o cadena de síntomas promedio se representa mediante un conjunto de  $n$  síntomas o variables, si estas variables son independientes entre si entonces cada objeto (imagen, regla o cadena de síntomas promedio) representado por las  $n$  variables se puede visualizar como un punto en un espacio de  $n$  dimensiones.



El objeto a se representa con dos variable independientes, por lo que se puede visualizar como un punto en 2 dimensiones



El objeto b se representa con tres variable independientes, por lo que se puede visualizar como un punto en 3 dimensiones

Si un objeto  $c$  se representa con cuatro variable independientes, entonces se puede visualizar como un punto en 4 dimensiones, y en general un objeto  $n$  que se representa con  $n$  variable independientes se puede visualizar como un punto en  $n$  dimensiones

O sea que cada vector de la matriz evolutiva representa un punto en un espacio multidimensional y todas las reglas forman una nube de puntos. De donde, por ejemplo, el proceso de encontrar el diagnostico asociado a un vector de síntomas  $X$  equivale a encontrar el punto que tiene distancia mínima con este vector.

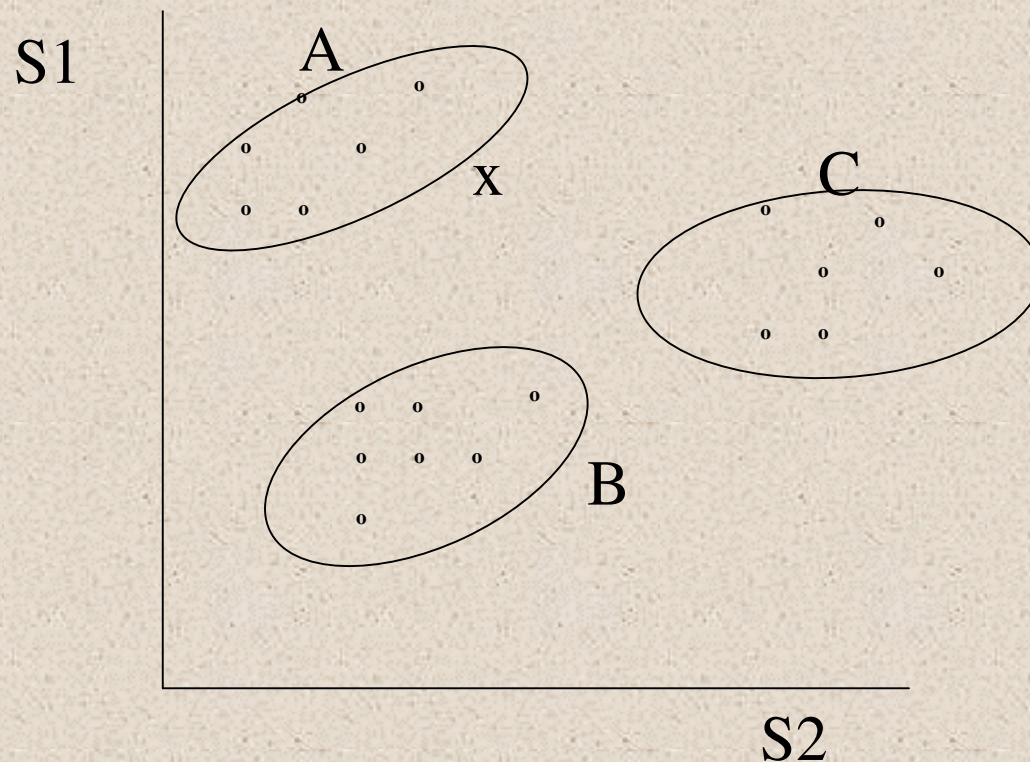
(Por facilidad visual mostraremos imágenes en 2 dimensiones , pero la idea se aplica a  $n$  dimensiones.)<sup>x</sup>



# **Espacios N-Dimensionales Evolutivos**

# Análisis de Cúmulos

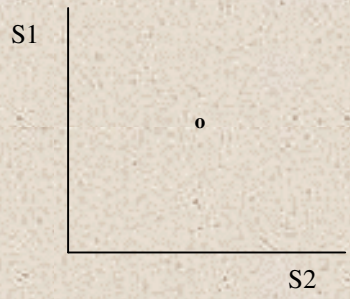
básicamente toma una muestra significativa de objetos similares, los representa como puntos en un espacio n-dimensional, encuentra la media y la varianza de estos puntos y cuando llega un objeto nuevo al sistema, mide la distancia entre el punto nuevo y el punto medio, si esta es pequeña lo reconoce como un objeto similar y si no lo desecha.

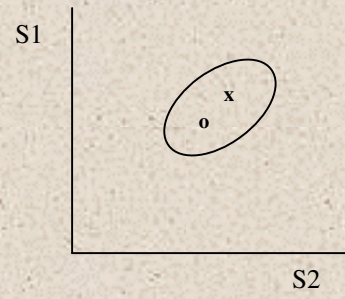
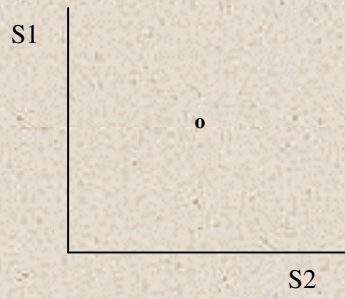


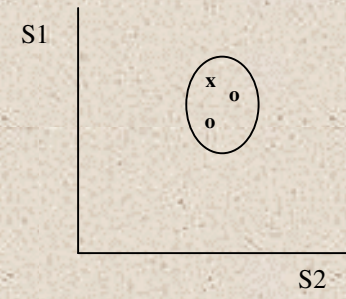
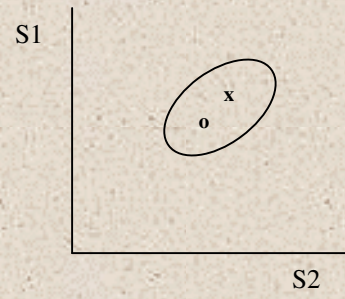
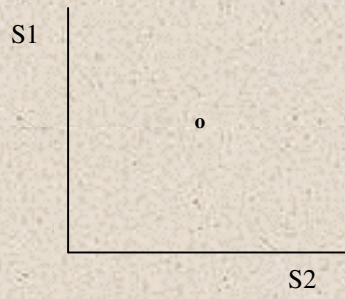
es común que cuando se aplica esta técnica, los patrones ya están prefijados, ya sea porque explícitamente fueron asignados o porque se obtuvieron mediante un proceso de aprendizaje y quedaron fijos e inmutables.

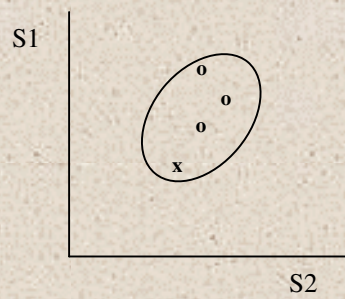
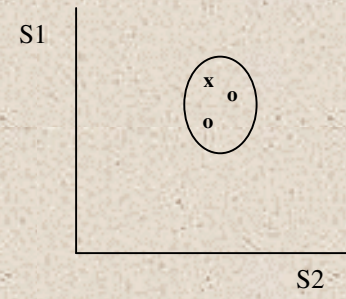
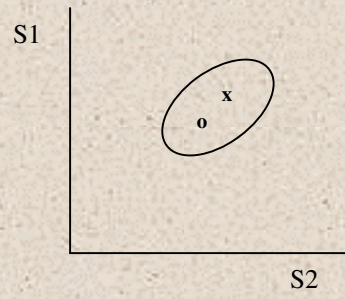
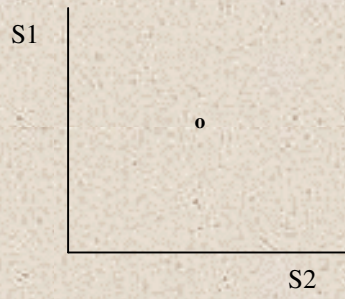
Ahora bien, en el caso de las matrices evolutivas también cada imagen, regla o cadena de síntomas promedio se representa como un punto en un espacio de  $n$  dimensiones, o sea que, todas las reglas forman una nube de puntos, pero con la característica de que *la matriz evolutiva permanentemente se esta modificando.*

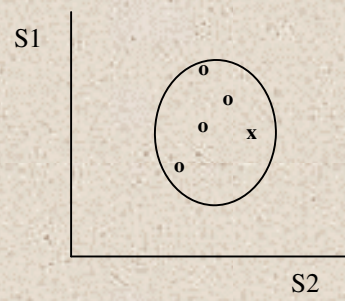
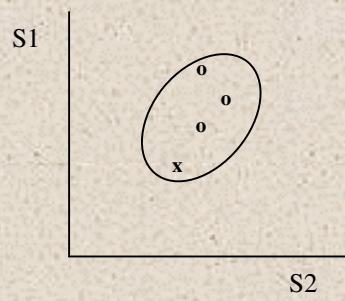
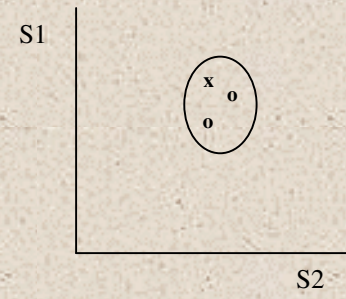
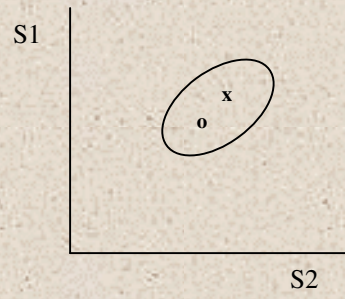
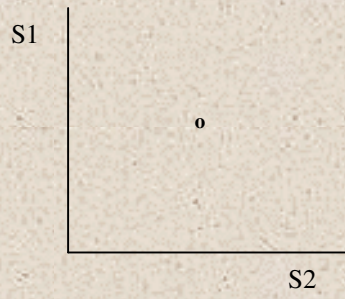
*Originalmente la matriz evolutiva esta vacía y representa un espacio que también lo esta, más adelante cuando llegan las primeras reglas o imágenes surgen los primeros puntos, pero estos no están fijos, cuando una regla de la matriz evolutiva se modifica el punto que la representa también cambia de posición, o sea que, en forma natural y permanente el espacio n-dimensional se esta afinando y evolucionando.*

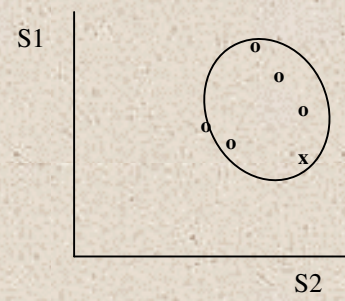
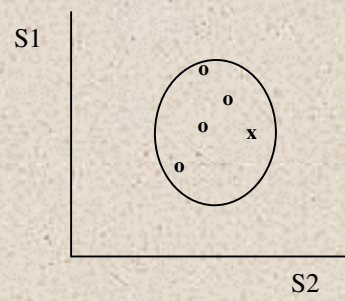
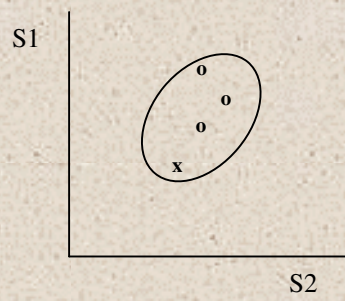
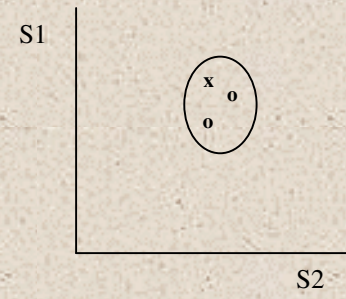
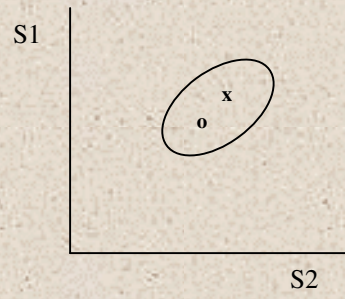
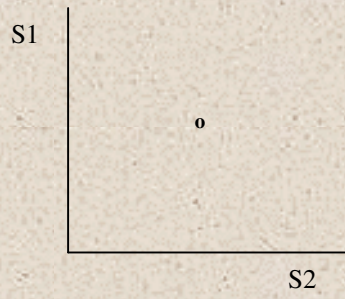




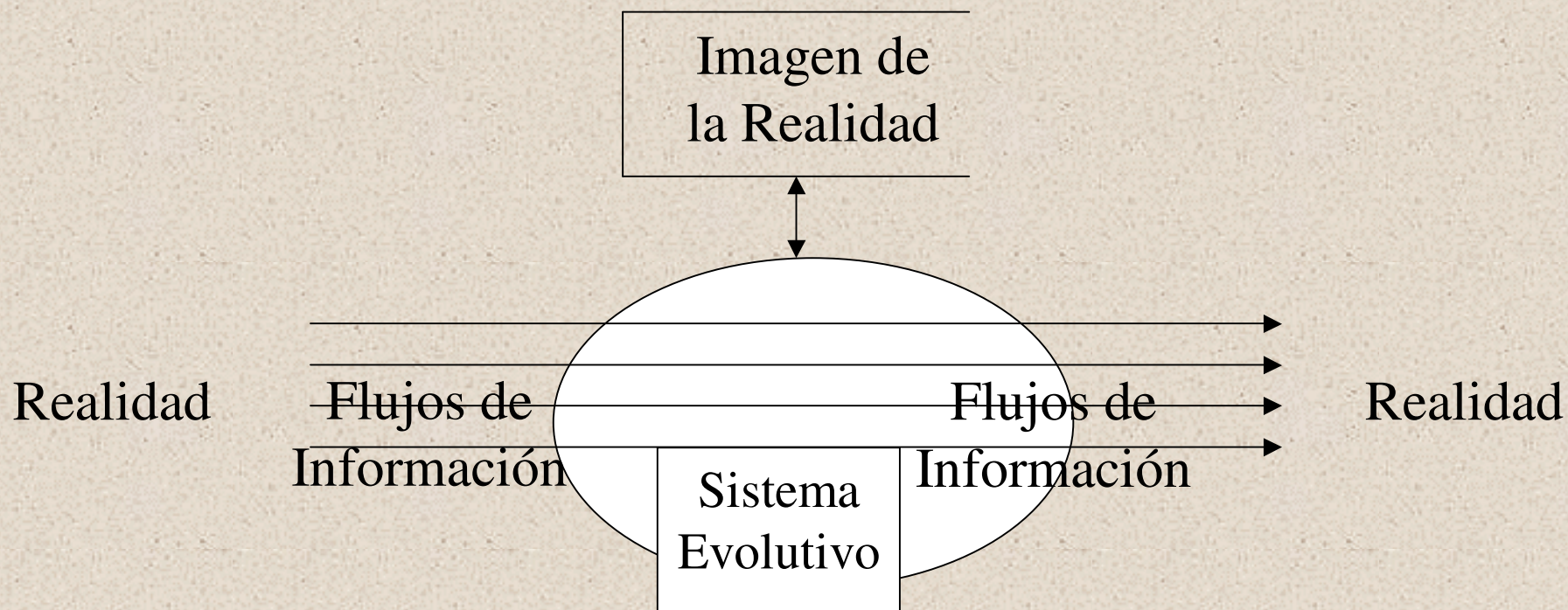








*Con lo que el sistema evolutivo está transformando permanentemente su imagen de la realidad y no existe un proceso previo de aprendizaje y otro de aplicación, sino que por el proceso natural de fluir la información, el sistema evoluciona.*

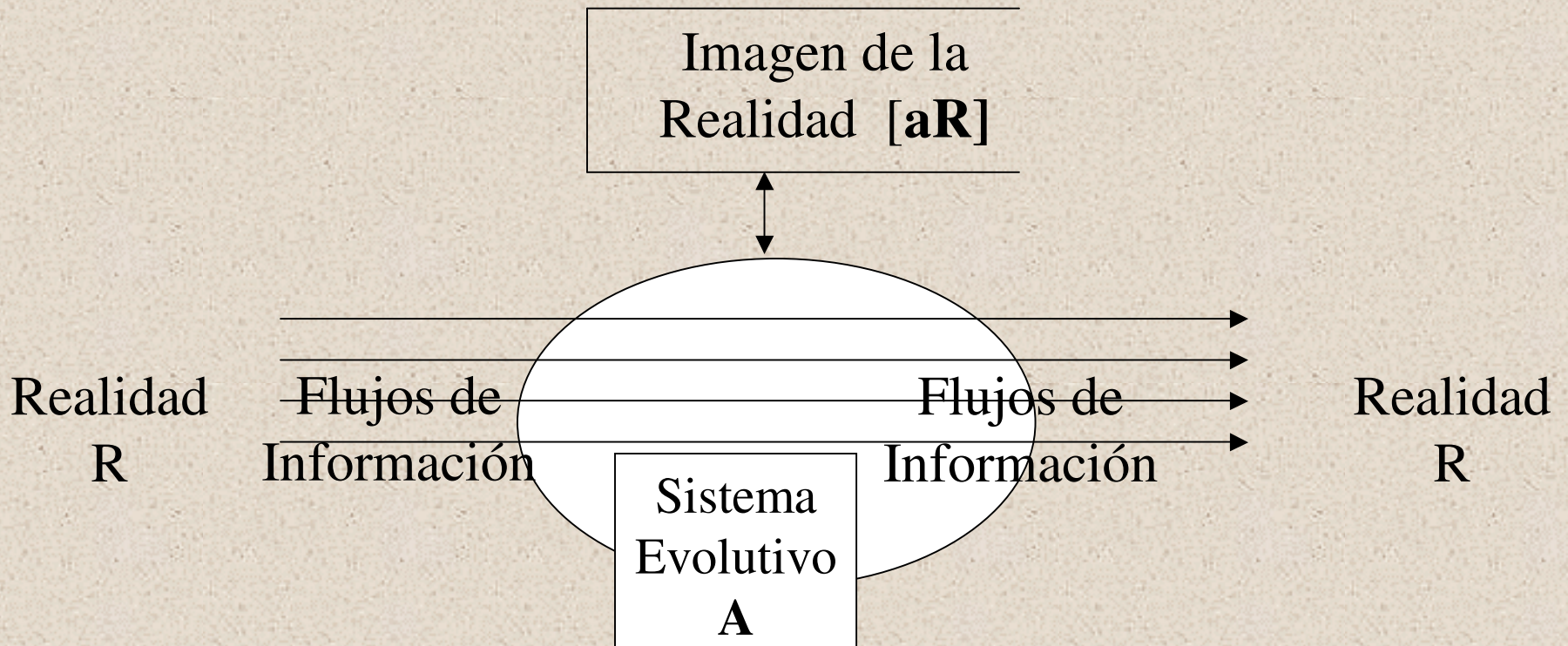


Por los flujos de información el Sistema Evolutivo se transforma

**Sistema Evolutivo 1993**

# Evolución de la imagen de la realidad

*Por el hecho de que fluye la información el sistema se modifica.*



Por los flujos de información el Sistema Evolutivo se transforma

**R** es la realidad, **A** es el sistema evolutivo

**A** percibe la realidad **R**, **aR**

**A** construye su imagen de la realidad **R**, **[aR]**

A construye su imagen de la realidad **R**, **[aR]**

Originalmente la imagen de la realidad **[aR]** esta vacía

Cuando empieza a fluir la información,  
el sistema empieza a percibir la realidad **aR**

y a construir la imagen de la realidad **[aR]**

**aR** → **[aR]**

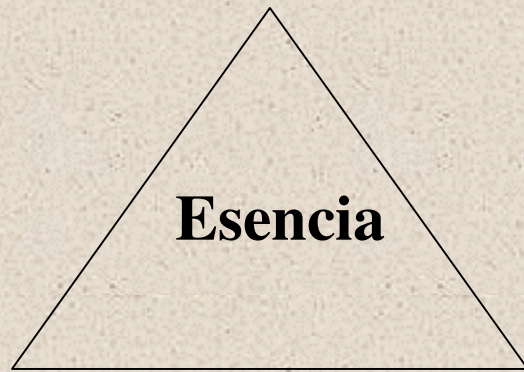
**El sistema permanentemente esta percibiendo la realidad aR**

**y su imagen de la realidad [aR] permanentemente esta evolucionando**

**[aR] ⊕ aR → [aR]**

# EVOLUCIÓN

**Información**



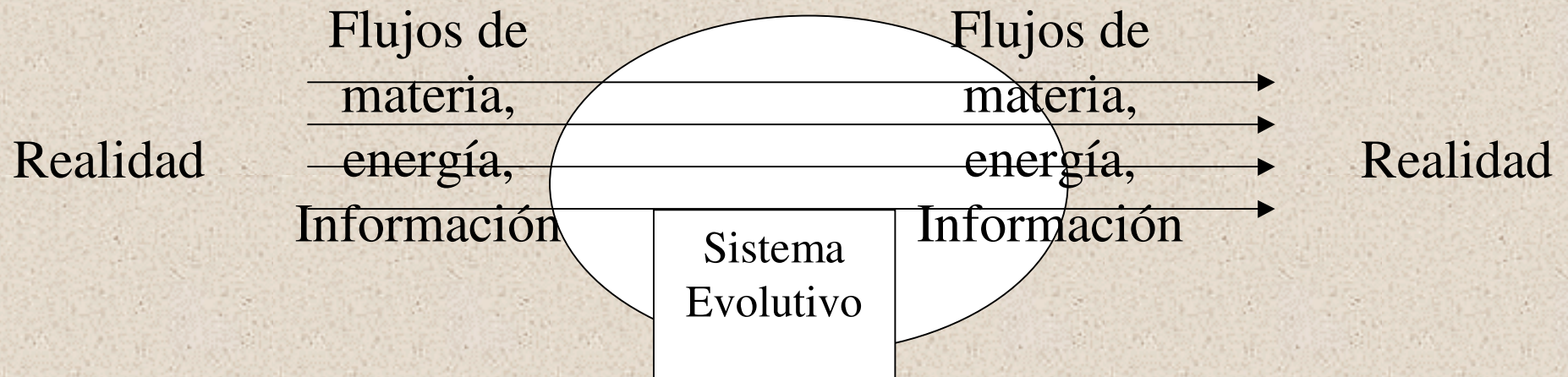
Generalizando a  
sistemas en los que fluye  
materia, energía e información

**Materia**

**Energía**

Factores esenciales

*Por el hecho de que fluyen la materia, energía, información o algunos otros factores esenciales se da la evolución.*



Por los flujos de materia, energía, información el Sistema evoluciona

**Evolución 1995**

Por ejemplo, por el hecho de que fluye a nuestro alrededor la información, nuestra imagen de la realidad cambia.

Un sistema evolutivo se puede visualizar como un paisaje con cordilleras, planicies y valles, cuando llueve el agua fluye por ese espacio y por el puro hecho de fluir hace que el espacio se modifique

*Ahora bien, el cambio se da como resultado de la interacción fractal que se da entre los componentes de un sistema, entre los sistemas, y entre todo esto y el enorme caldo de materia, energía e información donde se encuentra inmerso, ya que al haber interacción fluye algo entre los que interactúan y esto propicia la evolución.*

## Generalizando

llegamos a una serie de ideas generadas durante mas de veinte años en las que en esencia se plantea que

*la evolución, el crecimiento, la vida, el aprendizaje, el pensamiento, la transformación de nuestra imagen de la realidad, los procesos de descomposición, el desarrollo y transformación de las empresas, sociedades, organizaciones, países, galaxias y universos, etc., son manifestaciones de un mismo proceso general de transformación o cambio, y que existen reglas y propiedades generales que se aplican a las diferentes manifestaciones particulares.*

Por facilidad

*al concepto general lo denominaremos Evolución*

**Evolución 1995**

*al concepto general lo denominaremos  
Evolución,*

aunque lo podríamos llamar de muchas otras formas,  
como  
cambio o transformación.

O sea que,  
cuando nos refiramos a la evolución no nos estaremos  
refiriendo al concepto particular que tiene asociado,  
sino  
al concepto general con el cual integra y representa a todas las  
manifestaciones particulares.

Todos los organismos se mantienen en un *proceso permanente de evolución*, lo que pasa es que no lo notamos porque *en algunos casos es muy lento* (por ejemplo, los cambios naturales en las piedras y las estrellas tardan mucho en notarse) y *en otros muy rápidos* (una corriente de agua, una nube, etc.) y aun cuando la evolución se realice en tiempo real *o a una velocidad observable* (como la evolución de una empresa o idea, o del conocimiento que tenemos sobre un tema) *no somos conscientes de este proceso porque nadie nos lo ha hecho notar.*

*La Evolución es un fenómeno universal e intrínseco a la naturaleza y se presenta como resultado de la interacción de un organismo con su medio, con otros organismos, consigo mismo o cuando diferentes organismos y sociedades interactúan mutuamente en procesos de coevolución, dando como resultado que la organización o sistema (natural, artificial, social, etc.) se transforme y en su caso modifique el ambiente.*

# CONCLUSIÓN

En este trabajo se planteo que *múltiples manifestaciones de la realidad como la vida, la evolución, el aprendizaje, la transformación del universo y muchas otras, son casos particulares de una manifestación general, la cual esta regida por la interacción fractal de múltiples sistemas en un caldo en el que fluye permanentemente al menos la materia, energía e información, propiciando con este flujo los procesos de transformación.*

Con este trabajo espero dar una idea de la magnitud del área y propiciar que se puede empezar a plantear el surgimiento de una *Ciencia o Disciplina de la Evolución, Interacción o Cambio*, donde se pueda *estudiar la Evolución en sus diferentes manifestaciones y encontrar sus características (fundamentos, reglas, leyes, patrones, etc.) generales e independientes de las manifestaciones particulares en que se presenten.*