

# **CAPÍTULO 11**

## **CONCLUSIONES**

## 11 CONCLUSIONES

### 11.1 Introducción

En este capítulo se presentan las conclusiones generales y específicas de la presente investigación. Para ello se toma como referente las hipótesis de partida para contrastarlas con los resultados obtenidos y determinar si éstas se han verificado o no, sin dejar de lado los matices que se han podido encontrar a lo largo de la investigación. Para este propósito el capítulo se ha dividido en cinco secciones principales.

Se comienza, en la primera de ellas, por recoger las principales conclusiones propuestas en cada capítulo, para articularlas entre sí demostrando la coherencia del proceso desarrollado y para verlas en su globalidad. Obviamente, el énfasis de esta primera parte se hará en los resultados de la fase experimental como elemento articulador entre la exploración teórica y el desarrollo del software propuesto.

La segunda parte se dedica específicamente a la presentación de los elementos que le dan estructura y que definen la arquitectura básica de un software cuyo objetivo sea el asistir al ingeniero de diseño en aquella fase crucial del desarrollo conceptual de soluciones a problemas de diseño o desarrollo de nuevos productos.

En la tercera sección se hace el balance, antes mencionado, entre las hipótesis iniciales de la investigación y los resultados concretos obtenidos. Para ello se presenta el enunciado de cada una de las hipótesis para, entonces, presentar la argumentación sustentada en los resultados y pasar así a concluir sobre su validación.

En la cuarta sección se presenta la conclusión final de la investigación, basada en sus objetivos generales y específicos. Finalmente, en la quinta sección se dan algunas sugerencias sobre investigaciones y desarrollos futuros, que darían continuidad a esta línea de desarrollo.

### 11.2 Conclusiones de la exploración teórica y de la experimentación

#### 11.2.1 De la exploración teórica

##### *a. La complejidad del diseño*

Resulta evidente tanto por la revisión de la literatura sobre el tema de diseño como por la verificación experimental, que el proceso de diseño, entendido en su amplio significado,

está caracterizado por su complejidad causada por las múltiples interacciones entre el individuo, su cognición, sus experiencias, sus habilidades y sus conocimientos técnicos; el entorno social al que pertenece, cada vez más difuso debido al desarrollo de los medios de comunicación; y por la ciencia y sus leyes, que marcan fronteras que también son cada vez más flexibles debido a su gran dinámica de cambio.

La proliferación de métodos y modelos en los últimos cincuenta años es una muestra del interés creciente en la investigación en diseño. De ellos, el modelo de razonamiento en el ciclo de diseño (Takeda et al., 1990) parece el más adecuado para representar el proceso al integrar conceptos valiosos como el de la evolución (Chakrabarti, 2001; Maher, 2001), la abducción y la evaluación convergente.

Aun así, la modelación del proceso real de diseño se aprecia como un reto aún sin alcanzar, y todos los modelos propuestos, se puede, decir que son incompletos, incluyendo aquellos utilizados en esta investigación para representar y valorar el proceso de diseño bajo la perspectiva de la creatividad.

#### ***b. Vigencia del modelo Wallas de creatividad***

La misma característica de complejidad del diseño, acompaña al concepto de creatividad. Aun para definirla existen muchas propuestas. Aquí se ha adoptado una simple pero significativa para lograr asociarla al diseño de productos, aquella que la describe en términos de sus dos características relevantes e inseparables: novedad y utilidad.

A pesar de la existencia de varias propuestas para representar el proceso creativo, se puede concluir que el viejo modelo de Wallas (1926) de cuatro etapas básicas (preparación, incubación, iluminación y verificación) aún sigue teniendo vigencia. Otros modelos desarrollan más o le dan mayor importancia a alguna(s) de aquellas etapas, pero el concepto mismo de la presencia de cada una de ellas subyace como estructura fundamental.

De las tendencias o posiciones frente a la utilización del ordenador en actividades que demanden creatividad (como el diseño), en esta investigación se ha adoptado la intermedia, ésta es, la que afirma la utilidad del ordenador pero solo como herramienta a disposición del individuo que diseña y no como instrumento creativo por sí mismo. Esta conclusión se sustenta en el hecho de que, si bien es cierto, la literatura reporta varios desarrollos orientados a aquél fin, su estado actual es incipiente y poco útil para atender las necesidades reales a corto y medio plazo.

### **c. La conceptualización, fase olvidada de las herramientas CAD**

La historia del desarrollo de los programas CAD, los ubican como eje propulsor de las herramientas informáticas de asistencia al diseño, hasta el punto de haberse convertido en instrumento estratégico a nivel productivo industrial con altos índices de utilización. Su evolución sugiere que no es correcto catalogar a este tipo de programas exclusivamente como trazadores de gráficos, como cuando se originaron, ya que se han constituido como verdadera plataforma de integración de herramientas (trabajo colaborativo, análisis por elementos finitos, listas de materiales, etc.).

Sin embargo, por la revisión de los productos ofrecidos por las principales compañías desarrolladoras de este tipo de software, se puede afirmar que todas se centran en las fases de diseño de detalle (análisis) y que están enfocadas al diseño de partes antes que de ensambles, por lo cuál la etapa de síntesis no es soportada por este tipo de programas. La conceptualización es, por lo tanto, la fase olvidada de los programas CAD.

### **d. La inexistencia de software creativo para actividades técnicas**

Con excepción del software que desarrollan la metodología TRIZ, los programas de ordenador que se ofrecen comercialmente y que implementan técnicas creativas no han sido concebidos para aplicaciones específicamente técnicas, sino que tienen un carácter genérico, supuestamente aplicables a cualquier tipo de problema.

No existe ninguna evaluación objetiva y comparativa de tales tipos de programas y los reportes bibliográficos se limitan a describir la funcionalidad con fines comerciales.

Los programas del tipo TRIZ son costosos y complejos. Además su característica no es precisamente el estímulo creativo sino la búsqueda sistemática de soluciones técnicas, siguiendo pautas preestablecidas, casi algorítmicas.

No existe ningún programa de este tipo desarrollado para aplicaciones en países de habla castellana. Todos están en inglés y ello dificulta la aplicabilidad en contextos como los de España y Latinoamérica, máxime teniendo en cuenta el amplio uso de elementos del lenguaje como forma de interacción con el usuario.

### **e. La dificultad de medir la creatividad y lo creativo**

La medición de la creatividad está también en discusión. No existe una única escala ni mucho menos una única forma de valoración. Es más, puede tener hasta tres

dimensiones diferentes (la persona, el producto y el proceso) y cada una con sus propias dificultades.

Para medir la creatividad del individuo se seleccionó el modelo CREA, propuesto por Corbalán (2003), por tres razones: la argumentación clara sobre la que se sustenta, basada en la capacidad de formulación de preguntas; su sencillez para aplicarlo, por lo cual no fue necesario recurrir a análisis psicológico especializado; y la construcción y validación estadística del baremo, realizada con población española y latinoamericana.

Para medir la creatividad del proceso se revisaron varios modelos, pero no se encontró alguno que reuniera todas las características necesarias para los objetivos de esta investigación. Por ello se hizo necesaria una adecuación partiendo de la propuesta de Shah y Vargas (2003) porque, con los ajustes pertinentes, hizo factible la incorporación de las cuatro grandes características de la creatividad reconocidas por los más importantes investigadores del área: novedad, fluidez, elaboración y flexibilidad.

## **11.2.2 De los modelos y del método experimental**

### **a. La utilidad del estudio exploratorio**

La definición de las características de un experimento donde intervienen factores tan diversos y con interacciones tan complejas como el realizado en esta investigación, fue facilitada por la realización de un estudio previo de tipo exploratorio orientado, más que a la medición, a la observación de tales interacciones y a la identificación de los factores a tener en cuenta posteriormente. De las conclusiones del estudio exploratorio vale la pena citar:

- La necesidad de una capacitación específica en el uso del software, que fuese más allá del manejo formal del programa, enfatizando en los objetivos de cada módulo o cada técnica creativa ofrecida.
- La conveniencia de realizar el experimento en forma individual y no en grupo. Debido a que el objetivo es valorar la influencia del software en el proceso y resultados del diseño, resultó necesario mantener otros factores controlados, como por ejemplo, la influencia de líderes sobre el grupo, la pérdida de motivación por falta de atención de los otros miembros, la posibilidad de distracción, etc.
- La necesidad de permitir el uso de lápiz y papel como medios alternativos de expresión de ideas, debido a las dificultades inherentes al software para elaborar dibujos y esquemas

- La importancia de no depender exclusivamente de las expresiones verbales para la captura de la información relevante del proceso. Esto significó la necesidad de trabajar con un estudio de protocolo modificado que permitiese la captura de información verbal, visual y de interfaz del ordenador.

#### ***b. Alcances y limitaciones del método experimental***

El método experimental de «protocolo de diseño» es el más utilizado actualmente en investigación del diseño (Baya, 1996; Cross, 1996, Mulet, Vidal y Gómez-Senent, 2002; Tang y Gero, 2002). Es un método que busca capturar, en condiciones controladas, la interacción del diseñador durante el mismo proceso de desarrollo de su actividad, con lo cual es factible determinar las variables y los comportamientos con el fin de analizarlas posteriormente y llegar a conclusiones objetivas.

Una limitación importante del método es la imposibilidad de capturar toda la complejidad del proceso, especialmente aquellas actividades cognitivas que no son exteriorizadas (escrita, gestual o verbalmente) por el diseñador. Por ello la aplicación del método de estudio de protocolo permitió recoger solamente los siguientes aspectos de la actividad de diseño: palabras, gestos, manipulación de objetos, acciones realizadas en el software y dibujos trazados sobre papel.

Aunque limitada, aquella información capturada permitió la reconstrucción posterior de todo el proceso desarrollado, momento a momento, con lo cual se pudo valorar las diferentes características observadas en cada caso, para poderlas comparar y obtener los resultados buscados.

El procesamiento de la información registrada estuvo caracterizado por su laboriosidad, requiriéndose por cada sesión experimental, que tuvo originalmente un promedio de dos horas de duración, entre una y dos semanas de trabajo continuo para digitalizar, sincronizar, codificar, aplicar modelo de análisis, hacer ajustes, etc. Esto significó mucho tiempo para lograr terminar el procesamiento de las 25 sesiones realizadas.

El método no permite el control de efectos indeseados tales como la tolerancia de los diferentes sujetos a la tensión que significa ser grabados en vídeo, la presión del tiempo limitado, la respuesta al ambiente del lugar de grabación y el estado emocional, entre otros. Por ello el alcance de la investigación está supeditado a los factores externos visibles y «capturables» del proceso, mientras que aquellos otros se consideraron como factores no controlables del experimento.

### **c. El problema asignado**

El problema asignado a los participantes del experimento, denominado «plegadora casera de envases de tetrabrick», fue apropiado para los propósitos perseguidos. La valoración realizada por los propios participantes así lo ratifica, aunque hay que mencionar que en dos casos fue menos retador de lo esperado, de manera que para ellos el estímulo por desarrollar toda la sesión tal como esta prevista no fue suficiente y terminaron antes de tiempo.

### **d. El modelo de análisis de resultados**

Se ha comprobado la validez del modelo FBS de Umeda (1990) para representar el proceso de diseño. El uso de funciones, identificadas en el protocolo por los verbos; de modificadores, identificados por adverbios y adjetivos; y de estructuras de solución, por nombres; sirvió de base para la reconstrucción organizada del proceso y para la cuantificación de los resultados.

El modelo, sin embargo, tiene la dificultad de que la representación gráfica de la evolución del proceso resulta demasiado compleja para su comprensión. Evidentemente, quien construye tal representación la comprende, pero para el que se acerca como observador le resulta poco útil. Por esa razón en esta investigación se desistió de hacer la representación gráfica del proceso utilizando tal modelo y se prefirió utilizar los elementos identificados para procesarlos numéricamente y a partir de allí elaborar tablas de análisis y gráficas de comportamiento.

Como un aporte de esta investigación se ha propuesto aquella forma de procesamiento descriptivo-numérica presentada con detalle en el capítulo 7, construida con base en la identificación de elementos FBS y del concepto de interrelaciones propuesto por Goldschmith (1996). De esta manera se hace posible la identificación momento a momento del tipo de elemento (función, modificador, estructura), de las interdependencias entre ellos, de su grado de validez y del estímulo que lo origina.

Este nuevo modelo de representación permitió realizar el análisis durante cada momento del proceso de diseño y no solo de los resultados finales, lo cual era una condición necesaria para poder valorar la eficiencia de los diferentes módulos creativos propuestos en cada programa estudiado.

Otro aporte a destacar fue la revisión exhaustiva del modelo de Redelinghuys (1997a, 1997b), en un principio considerado como adecuado para el estudio, pero finalmente

descartado debido a sus limitaciones, en particular, por considerar que aún es necesario validarlo en forma efectiva en la práctica.

Se introduciendo una versión modificada de la propuesta de Shah y Vargas (2003), de tal manera que se pueda hacer una la valoración de la creatividad no solo de los productos finales sino de los que se podrían denominar como productos intermedios, éstos son, aquellos que se van generando durante el proceso aun cuando no maduren hacia productos finales. En capítulo 7 se explican los detalles, pero aquí se puede afirmar que se logró la cuantificación a través de índices de la «calidad creativa» del proceso y no solo del producto. Este aporte puede considerarse como significativo para estudios cuyo objetivo sea comprender el proceso y cuantificar resultados intermedios. Un aporte específico de la investigación es el procedimiento para valorar los índices de fluidez y de elaboración de las ideas que se generan durante el proceso de diseño, que complementan a los de novedad y flexibilidad reportados en su forma básica por Shah y Vargas (2003).

De esta manera se puede señalar, como un aporte nuevo y valioso de esta investigación, la obtención de un modelo completo de valoración objetiva del proceso de diseño bajo la perspectiva de la calidad creativa a partir de protocolos no exclusivamente verbales, que posibilita la comparación objetiva entre los diferentes tipos de programas estudiados. Tal modelo podría, igualmente, ser útil en otros tipos de investigaciones en el área del diseño.

### **11.2.3 De los resultados experimentales**

#### ***a. Por programa individual***

- Axón Idea Processor©

El diseño estructural y de la interfaz del programa induce a su utilización solamente como editor de mapas mentales, aunque dispone adicionalmente de más herramientas estimuladoras de ideas. De hecho el 83% de las ideas generadas por el programa se originaron por el trabajo con mapas mentales.

La cantidad de opciones disponibles para editar los mapas mentales son poco útiles y causan distracción del usuario. Al contrario, destaca como muy útil la posibilidad de ampliar la explicación de cada elemento del mapa mental, por medio de las ventanas emergentes asociadas.

En cuanto a la cantidad de ideas generadas, el experimento revela un equilibrio entre el número de ideas cuyo origen detectado fue el propio programa y aquellas propiciadas por otras fuentes, lo cual es un indicador de que el programa, aun con sus deficiencias señaladas, es útil para ayudar en la generación de ideas.

En la medida que se avanza en el proceso, el ritmo de generación de ideas comienza a disminuir sensiblemente. Esta característica se podría considerar como normal, pero en este caso tal reducción es mucho más significativa. Se puede afirmar, por lo tanto, que la técnica de mapas mentales es útil principalmente como organizadora de ideas y estimula en la fase inicial del proceso, pero luego pierde eficacia.

- Brainstorming Toolbox©

Se aprecia un mayor equilibrio del uso de las diferentes opciones del programa, aunque el módulo «random picture» marcó diferencia, debido al hecho de que las figuras transmiten mayor información que las solas palabras y porque aquellas estaban escritas en inglés, con la dificultad añadida de usar términos poco comunes y frases hechas. El aspecto del idioma se muestra como altamente relevante sobre la efectividad de las herramientas dependientes de elementos del lenguaje no gráfico. Por estas mismas razones aquella herramienta fue la iniciadora del 50% de las ideas propuestas a partir del programa.

La característica más apreciada de este programa fue su sencillez. Ello facilita el proceso de aprendizaje y permite la concentración en la actividad propia de búsqueda de alternativas de resolución del problema en lugar de desviarla hacia elementos formales del software.

- Creax Innovation Suite©

Es el software con mayor grado de dificultad de aprendizaje y de utilización, debido a que se sustenta en la metodología TRIZ, de por sí compleja.

Se aprecia una tendencia a generar pocas ideas de alternativas de solución, pero con mayor grado de desarrollo, lo cual deja claro que el software tiene mayor orientación hacia fases de convergencia que a fases de divergencia.

Las dos herramientas más utilizadas fueron la búsqueda de recursos del sistema y la interfaz de construcción del modelo funcional. El primero porque conduce la atención hacia elementos que normalmente pasan desapercibidos y su descubrimiento genera expectativa de encontrar rutas de solución al problema. La segunda, porque al ser una

herramienta de tipo gráfico resulta más natural para el diseñador como medio de expresión, aunque en este caso, por el uso de iconos no estandarizados resulta un tanto incomodo su uso.

Un indicador de la relativa eficiencia de este software fue el hecho de que el 65% de las ideas generadas se originaron en alguno de sus módulos, frente al 35% proveniente de acciones externas al software. Esto a pesar de que la mayor parte de las herramientas que conforman el software no fueron utilizadas o lo fueron muy superficialmente. Eso sí, hay que matizar estas cifras diciendo que el promedio de ideas generadas en las sesiones que utilizaron este programa fue el más bajo de todos.

En particular resultó frustrante el tratar de utilizar herramientas como la matriz de contradicciones o las tendencias de la evolución, y no lograr establecer un puente que condujese a soluciones claras. Este resultado obedeció a la falta de un entrenamiento previo, sobre la metodología TRIZ, más intensivo y a que la interfaz propuesta por el programa no se caracteriza precisamente por su usabilidad.

- ThoughtPath Problem Solver™

Se trata de un programa de tipo «prescriptivo» que conduce al usuario a través de los diferentes módulos siguiendo un esquema más rígido que en los otros programas, lo cual se refleja en la cantidad de tiempo empleado utilizando el programa frente a la opción de trabajar con lápiz y papel.

Por esta misma razón, los diferentes módulos del programa muestran aproximadamente la misma intensidad de uso, representado tanto por el tiempo empleado como por el número de ideas asociadas a cada uno de ellos. Aquella característica facilita la utilización del programa ya que en todo momento está siendo guiado por una ruta específica, pero a su vez crea una sensación de que se está construyendo una solución válida cuando puede ser que en realidad no sea así.

La forma reiterada en que el programa recomienda el uso de los denominados «wishes» termina por cansar al usuario quien desea avanzar en forma efectiva a la solución. Por otro lado, los estimuladores de ideas utilizan con mucha frecuencia elementos complejos del lenguaje escrito como por ejemplo poemas, refranes y frases hechas, que al estar en inglés pierden parte de su potencialidad cuando un hispanohablante lo utiliza.

- Sin uso de programa

Resulta evidente la tendencia de generar un gran número de ideas de solución en los primeros momentos del proceso para luego ir decayendo paulatinamente sin mostrar ninguna forma posterior de recuperación significativa del flujo de ideas. Por ello la trayectoria creativa tiene el comportamiento singular, muy alto al inicio y luego una permanente disminución hasta el final.

De esta manera, una de las conclusiones fundamentales de la fase experimental es que cuando no existen estímulos externos, la fuente de ideas se limita al conocimiento y a la experiencia previa del diseñador; de ahí el comportamiento señalado. Mientras que el uso de software estimula la generación de ideas por asociación, transformación y exploración, en función de los diferentes tipos de estímulos provistos (metáforas, gráficas, frases, preguntas, listas de chequeo, mapas mentales, etc.).

En contraposición se puede concluir dos desventajas importantes del uso de software. En primer lugar, la necesidad de un entrenamiento previo y de un proceso de familiarización con su uso, lo cual puede requerir más o menos esfuerzo en función de las técnicas utilizadas, del diseño de interfaz y de la tolerancia del usuario. En segundo lugar, el hecho de que utilizar un software para este trabajo introduce un elemento, hasta cierto punto extraño o poco natural, que puede ocasionar la distracción del diseñador hacia actividades que no son indispensables en el proceso de conceptualización de soluciones (edición de texto, corrección de formas, adornos innecesarios, etc.).

### ***b. Resultados globales***

En primer lugar, se puede afirmar que la utilización de software requiere de parte del usuario un alto grado de atención que le impide concentrarse en otras actividades paralelas, como por ejemplo, expresar audiblemente sus ideas. En esta limitación incide el grado de dominio y de experiencia que se tenga en el manejo del software, de manera que cabe esperarse que en la medida que el dominio, la fluidez y la confianza en el manejo del software aumente con la práctica, esta limitación pierda importancia.

También se puede decir que el software no incide significativamente en las primeras etapas del proceso de desarrollo de productos. Es evidente que en casi todas las variables estudiadas el tratamiento sin software mostró mejor comportamiento durante los primeros diez minutos de la sesión. Esto se puede explicar por el hecho de que el software evaluado requiere que durante los primeros minutos de trabajo el usuario se centre en detalles formales (nombre del archivo, por ejemplo) antes de entrar en los módulos efectivos de trabajo. Es decir, el tiempo de preparación para la sesión es mucho

mayor en el caso de utilizar software que cuando no se utiliza. En la medida que el programa demanda más información preliminar o requiere más lectura de instrucciones, seguramente se hará menos atractivo para el usuario, normalmente poco tolerante al aprendizaje de software.

En la medida que avanza el proceso se va notando la incidencia positiva que el programa tiene sobre el usuario para generar ideas. De esta manera vale la pena resaltar el hecho de que es posible descartar los prejuicios establecidos sobre el uso de software como elemento que frena la generación de ideas, tal como quedó demostrado por el análisis de los resultados experimentales, tanto por la cantidad y el flujo de ideas, como por la calidad del proceso mismo de diseño.

El análisis del origen de las ideas permite ver una diferencia significativa entre los tratamientos que utilizaron software con respecto al que no utilizo. Evidentemente que en este último caso las fuentes de ideas son, principalmente, la experiencia y el conocimiento acumulados, con lo cual, el éxito de la innovación en un producto dependerá mucho de lo que propongan aquellos que han tenido un contacto previo y una buena preparación relacionada con el problema a resolver.

Al utilizar software se potencia la posibilidad de generar ideas a partir de estímulos que permiten recuperar información de la memoria de largo plazo para utilizarla mediante establecimiento de relaciones y por transformar o combinar conceptos. En este aspecto destacan los casos que utilizan algún medio de interrelación gráfico, tal como fotografías, que presentan gran cantidad de información en una sola pantalla. Es claro que la experiencia y los conocimientos del diseñador siguen teniendo protagonismo central y el software lo que hace es establecer un «puente» entre los estímulos externos y la mente del diseñador, con lo cual se pueden salvar los bloqueos mentales que se suelen presentar en condiciones normales de diseño.

También hay que afirmar que el software al disponer de elementos estimuladores de tipo genérico, es decir, elementos que no están concebidos para atender un tipo de producto o problema específicos, requiere alguna forma de catalizar o traducir aquellos estímulos al problema que se quiere resolver. Esto fue evidente por la mayor cantidad de tiempo que emplearon los participantes que utilizaron software en manipular los envases de tetrabrik®, en contraste con el tiempo que gastaron los que no utilizaron software. El manipular las cajas les permitió establecer esa especie de «traducción» de las ideas genéricas propuestas por el software a ideas concretas para solucionar el problema planteado. Parece, por lo tanto, que el tocar con las manos sirve para darle forma a aquellos conceptos que de otra manera se quedan en un nivel de abstracción tal que los

hace poco útiles. Esta última afirmación, sin embargo, debería ser verificada por metodologías cognitivas fuera del alcance de esta investigación.

Otro aspecto interesante es el relacionado con la rigidez o flexibilidad del software. El forzar un camino a seguir para resolver el problema puede traer dos consecuencias antagónicas. Por un lado, provee la estructura deseable por el ingeniero, que siempre esta dispuesto a percibir estructuras de solución. Por otro, restringe las posibilidades y preferencias propias del usuario. El caso típico aquí estudiado es el del ThoughtPath: dirige al usuario por un camino preestablecido, de tal manera que el tiempo empleado en el uso de las diferentes herramientas o módulos fue casi el mismo, en contraste con otros tratamientos donde se aprecia una clara preferencia por parte del usuario. Sin embargo, los resultados obtenidos con este software no fueron los mejores. Así, será importante para el diseño de un software eficaz, mantener un alto grado de flexibilidad para que el usuario tenga libertad de elección, pero también guiarlo para que no se desvíe demasiado en cosas que pueden conducir a la intrascendencia. Esto significa que debe haber un equilibrio entre la flexibilidad y la estructura.

En cuanto a la calidad del proceso de diseño, destaca el caso del índice de flexibilidad, sobre el cual, el utilizar o no software marcó diferencias significativas. Se puede afirmar que el trabajar en la manera tradicional (sin uso de software) implica una rigidez alta en las soluciones. Los participantes se centraron en unas pocas categorías de solución (principios físicos o de trabajo) y sobre ellas desarrollaron toda la sesión. Mientras que el uso de software permitió «ver» otras alternativas, otras rutas, que condujeran a una solución satisfactoria. Esta característica puede ser muy valorada a la hora de desarrollar productos innovadores.

Esta misma observación tiene vigencia para el caso de la fluidez de las ideas a través del proceso. Cuando se trabaja de la manera tradicional, sin software, la fluidez fue muy grande al inicio, pero luego decayó rápidamente y se mantuvo en niveles bajos durante todo el resto del proceso, mostrando así una tendencia predominantemente convergente. Mientras que el uso de software estimula ciclos de divergencia mucho más acentuados en toda la sesión, con lo cual se logra una mayor estabilidad en la fluidez de ideas mirada a lo largo de todo el proceso de desarrollo del producto.

La originalidad, como indicador especialmente relevante de la creatividad, muestra una tendencia a favor de dos de los programas estudiados, el Brainstorming Toolbox y el Axon Idea Processor, pero a su vez, los otros dos programas resultan pobremente valorados. Así pues, las características de aquellos dos deben ser muy tenidas en cuenta a la hora de proponer una herramienta informática integral: la sencillez, la flexibilidad y la

interacción de tipo gráfica. Por otra parte, de las dificultades relevantes de los otros dos programas, destacan la rigidez y la dificultad de aprendizaje y de ejecución.

### **11.3 Estructura del software propuesto**

Con referencia a la propuesta de software creativo orientado a las actividades de desarrollo de conceptos para nuevos productos y que fue presentada en detalle en el capítulo 10, se pueden inferir las siguientes conclusiones:

#### **11.3.1 Tiene en cuenta al usuario**

El software esta dirigido a ingenieros dedicados a actividades relacionadas con el desarrollo conceptual de nuevos productos y que tienen como lenguaje natural de comunicación el castellano.

La estructura del programa se realizó sobre la base de una revisión exhaustiva y un estudio experimental también exhaustivo, para determinar las características del comportamiento del usuario frente a un programa de este tipo. Por ello, se afirma aquí, que los criterios utilizados para la selección de la estructura y las diferentes opciones, se han realizado sobre una base sólida de conocimiento del tipo de usuario del programa.

El lenguaje utilizado es común por este tipo de usuario y se evita el uso de terminología extraña o ambigua. Solamente se ha permitido el acceso a fuentes externas que manejan el idioma inglés en aquellos casos considerados como imprescindibles (bases de datos de patentes, por ejemplo).

La iconografía utilizada para representar los diferentes elementos del software es la estandarizada por el uso frecuente en el mundo de la informática. Cuando se requirieron nuevos iconos se seleccionaron aquellos que evocan la tarea asignada.

#### **11.3.2 Utiliza un modelo estructural coherente**

Se seleccionó como modelo estructural el llamado FBS propuesto por Umeda (1990). Con él se logra una articulación coherente con el proceso de diseño y una forma de representación de tipo gráfico que se asimila en su uso a los mapas mentales, y en la que la función se constituye en el elemento importante, de manera que el diseñador parta de su identificación (de la función) y se dirija a la definición de estructuras que la desarrollen

(estructuras de solución), evaluadas por medio de su comparación con los requerimientos y especificaciones (modificadores funcionales).

La estructura orienta el proceso pero no restringe al usuario. Con ello se pretende lograr un balance adecuado entre estructura y flexibilidad dándole así una característica que se podría denominar «flexibilidad orientada», que dé libertad, pero que conduzca a un destino, que oriente pero no restrinja,.

### **11.3.3 Progresión de uso**

El proceso es orientado a través de tres fases interrelacionadas y retroalimentadas. En la primera se identifica el sistema-problema, esto es, la situación que se quiere resolver pero no vista como algo que sucede en forma aislada, sino perteneciente a un sistema que ofrece recursos y que impone restricciones. Tal sistema se busca que sea analizado desde varios puntos de vista, bajo varias perspectivas.

En la segunda fase se trabaja en la búsqueda del «resultado final ideal» con lo cual se pretende propiciar «saltos» tecnológicos que conduzcan a innovaciones importantes. Es decir, se trata de evitar que el proceso se limite solamente a proponer mejoras del producto y más bien se estimula a la búsqueda de nuevas soluciones. Para ello el programa ofrece herramientas creativas y bases de datos que el usuario puede utilizar a voluntad.

En la tercera fase se desarrolla la llamada aquí «evolución funcional controlada», en la que se van adicionando funciones y haciendo más complejo el modelo de solución, pero siempre en forma controlada mediante la evaluación en cada ciclo de evolución. Es decir, se busca que el usuario no adicione funciones por adicionarlas solamente, sino que tal adición obedezca a la necesidad de hacerlo por la evaluación del estadio anterior del sistema de solución; lo que Takeda (1990, 2001), explicando el proceso de síntesis, denomina como acciones cognitivas de deducción-abducción.

### **11.3.4 «Modularidad»**

Una de las características relevantes de la propuesta es el desarrollo del concepto de módulos que pueden incorporarse como herramientas del programa en función de su disponibilidad y de las preferencias del usuario. Así, el programa tiene una estructura básica funcional, alrededor de la cual se pueden agregar módulos de técnicas y herramientas creativas y de manejo de conocimiento.

Esta característica tiene implicaciones tanto para el desarrollo del software como para su utilización. Para el desarrollo porque significa la posibilidad de abordarlo por etapas secuenciales cuyos resultados se adicionarán a la estructura funcional básica. Y para la utilización, porque el usuario podrá seleccionar a voluntad los módulos que más se ajusten a sus necesidades y preferencias, sin perder funcionalidad.

### **11.3.5 Incorpora módulos de técnicas y herramientas creativas**

Las técnicas y herramientas seleccionadas fueron las más representativas según el análisis experimental. De tal manera que las que se incluyen han sido valoradas objetivamente como útiles para ayudar a la generación de alternativas de solución. Estas son: Figuras aleatorias, Palabras aleatorias, Listas de chequeo, SCAMPER, Juegos de rol, Preguntas clave y Manejo de contradicciones.

Son presentadas como módulos que se incorporan al software dentro de una barra de selección, de manera que pueden ser utilizadas en cualquier momento sin tener que seguirse un orden preestablecido. El diseño de cada una de ellas tiene también en cuenta los comentarios y valoraciones obtenidas durante el experimento y el análisis de resultados.

### **11.3.6 Ofrece módulos de herramientas de gestión de conocimiento**

Las cuatro clases de bases de datos que integran el software ofrecen un abanico importante para el almacenamiento y recuperación de información técnica relevante para un ingeniero de diseño. Además permite la construcción de una «memoria técnica de diseño» que posibilita el almacenamiento de ideas y de procesos de solución que son potencialmente útiles para la empresa cuando aborde diferentes tipos de proyectos.

Como en el caso anterior, estas bases son módulos independientes, agregables en función de las necesidades y preferencias del usuario. Para su implementación se sugiere la utilización de un sistema de acceso y filtrado por medio de alguna herramienta de alto nivel, que facilite la búsqueda de la información realmente pertinente. Para ello se presentaron dos alternativas: sistema de filtrado por gestión semántica automática, búsqueda delegada en web-robots.

## 11.4 Verificación de hipótesis

Aquí se presenta la comparación entre las cuatro hipótesis de partida de esta investigación y los resultados encontrados después de realizarla. Para cada una de ellas se presenta la argumentación que permitirá deducir su aceptación o su rechazo. Estas son:

- **Hipótesis de posibilidad.** *La generación de ideas dentro de la etapa de conceptualización del proceso de diseño en ingeniería puede ser asistida por ordenador, de manera que se obtengan soluciones novedosas y apropiadas. Su utilización permitirá explorar y ampliar el espacio de diseño, de manera que se genere mayor número de ideas y de mejor calidad.*

El uso de estímulos adecuados facilita la exploración en el espacio de diseño brindando nuevas alternativas de solución. La evaluación del software comercial muestra su efectividad, aunque también deja al descubierto sus debilidades, lo cual se capitaliza en la propuesta de un nuevo programa que reúna las mejores características para los efectos comentados.

El número de ideas generadas por utilizar un software no fue mayor que el obtenido sin utilizarlo, pero la calidad del proceso fue mejor y más consistente a través del tiempo, principalmente debido a que mantiene durante todo el proceso abierta la posibilidad de nuevas exploraciones y búsquedas, mediante la aplicación de las diversas técnicas provistas. Al contrario, cuando se trabaja sin software las soluciones se suelen buscar en un espacio de diseño restringido a la experiencia y conocimientos propios del diseñador, por lo cual los principios físicos que se aplican son muy limitados.

Los resultados encontrados permiten afirmar que efectivamente la producción de ideas conceptuales de solución de problemas de diseño puede ser potenciada por la mediación de un software apropiado, verificando de esta manera el cumplimiento de la primera hipótesis.

- **Hipótesis de efectividad.** *La efectividad práctica de tal tipo de programa puede ser evaluada mediante la definición o adopción de un modelo para la medición del grado de novedad y viabilidad del producto diseñado.*

La literatura especializada reporta varios modelos propuestos para estudiar tanto el proceso de diseño como el proceso creativo. Sin embargo, como tales modelos no se

adecuaban a las características propias de esta investigación, hubo necesidad de tomar elementos de varios de ellos como base para construir un modelo apropiado, presentado en el capítulo siete. Tal modelo se verificó mediante la aplicación al estudio experimental realizado, de tal manera que ha sido también validado, pudiéndose afirmar que los resultados obtenidos mediante su aplicación permiten varios tipos de análisis del proceso de diseño y no solo del producto diseñado.

Se pudo evaluar las siguientes características del proceso de desarrollo de productos: tiempo utilizado para cada actividad (tanto las desarrolladas dentro del propio software como externas a él), flujo de ideas, ideas totales producidas por cada actividad, origen de las ideas y creatividad del proceso valorada a través de índices cuantificados de la flexibilidad, fluidez, elaboración y novedad. Con estas características valoradas objetivamente es posible, como se demostró, la comparación entre las diferentes opciones que puede utilizar un ingeniero de diseño como herramientas de asistencia al proceso de desarrollo de nuevos productos.

Por lo tanto, la validez de esta hipótesis quedó demostrada por la construcción, validación y ajustes de un modelo para la valoración del proceso de diseño de nuevos productos con los criterios aceptados de la creatividad.

- **Hipótesis de estructura.** *Un programa de ordenador orientado a apoyar la etapa de generación de ideas conceptuales en ingeniería de diseño debe incluir al menos tres componentes centrales: estímulo para la generación y organización de ideas, bases de datos de conocimientos y diseño adecuado de la interfaz. Deberá orientar el proceso de diseño, pero de ninguna manera forzar o restringir la manera particular que el usuario tenga de abordar el problema. Para ello debe tener un alto grado de flexibilidad.*

Efectivamente, la comparación del comportamiento y de la efectividad de los cuatro tipos de programas evaluados, permitieron comprobar la importancia que tiene el contar con técnicas y herramientas que faciliten o establezcan un puente que lleven a generar ideas conceptuales nuevas. La ejecución de esta fase temprana del proceso de diseño con los medios convencionales (lápiz y papel) muestra una explosión inicial de ideas, pero a partir de ahí las posibilidades de encontrar nuevas soluciones se reduce drásticamente y el diseñador se concentra solamente en aquellas ideas o en sus principios físicos que representan. Mientras que cuando utilizan herramientas de estímulo creativo como las provistas en los programas, aquella generación de ideas se mantiene a lo largo del proceso, de manera que los ciclos de divergencia son mucho más evidentes.

Una de las deficiencias importantes encontradas en los programas evaluados fue la carencia de medios adecuados para establecer una «traducción» de las ideas que se generan a su aplicación concreta al problema que se resuelve. Esto generó la necesidad de manipular los envases de tretrabrik, buscando con ello establecer ese puente comentado. La necesidad de «pensar con las manos» parece difícil de obviar por medio de software; sin embargo, el ofrecer fuentes de información técnica concreta y relevante es una ayuda valiosa para contrastar las ideas que se van generando, de tal manera que las bases de datos de conocimiento cumplen un papel importante en los ciclos de convergencia del proceso. Por ello en la estructura propuesta para el software se incluyen paralelamente a las herramientas creativas, una serie de bases de datos que ofrecen aquel tipo de información útil y oportuna.

La usabilidad fue otro de los criterios básicos para la propuesta de software presentada en esta investigación. Por la observación de las sesiones experimentales, las preferencias manifestadas por los usuarios y la cantidad y calidad de las ideas generadas, es posible afirmar la existencia de errores del concepto de interfaz de los programas estudiados. Por ello, es clara la necesidad de utilizar criterios básicos de usabilidad para que la interfaz conduzca al efectivo aprovechamiento de todos los elementos del software.

Estas conclusiones sustentadas en los resultados de la fase experimental presentados en los capítulos ocho y nueve, y en la propuesta del nuevo software que se presentó en el capítulo diez, permiten concluir la validez de esta hipótesis de partida, aunque esta afirmación queda matizada por lo comentado sobre la necesidad de «pensar con las manos».

- **Hipótesis del método.** *A partir de evaluaciones experimentales de software creativo aplicado a la resolución de problemas de diseño, de la experiencia, de los conocimientos y métodos de trabajo de ingenieros de diseño y de los resultados de investigaciones recientes sobre diseño, creatividad técnica y gestión del conocimiento, es posible identificar elementos relevantes y características particulares de los tres componentes mencionados en la hipótesis anterior, así como técnicas y métodos creativos, que debe contener un programa de ordenador para potenciar la generación creativa de ideas y para que pueda utilizarse con más efectividad y eficiencia en ingeniería de diseño.*

El método de investigación empleado contempló cada uno de los elementos señalados en esta hipótesis, aunque su eje pivote fue la evaluación experimental. Por sus resultados y posterior análisis, se han deducido los principales componentes del

software propuesto, sobre la base de la efectividad lograda por cada una de las técnicas provistas en los diferentes programas evaluados.

La revisión exhaustiva en la literatura especializada sobre modelos de diseño y de creatividad, también aportó conocimiento necesario para poder estructurar el software cuya arquitectura básica aquí se ha propuesto.

Se puede concluir, por lo tanto, que la hipótesis ha sido válida. No obstante, es necesario mencionar la limitación del alcance de esta investigación al no incluir criterios complementarios que podrían haber enriquecido la propuesta final, como por ejemplo, conceptos de diseño «colaborativo» en red y elementos cognitivos en diseño.

## **11.5 Conclusión final**

Se ha logrado cumplir con los objetivos propuestos al iniciar esta investigación. La definición de la estructura de básica y la arquitectura de un software que asista al ingeniero de diseño durante el proceso de generación de ideas creativas en la fase de conceptualización de nuevos productos, ha sido lograda con base en un estudio experimental detallado y crítico de las alternativas ofrecidas actualmente. De esta manera se puede afirmar que el software propuesto como resultado final, no obedece a criterios comerciales o personales que buscan defender una metodología o una técnica creativa específica, sino que se sustenta en resultados objetivamente obtenidos.

El resultado final de la investigación presentado en forma concreta en el capítulo diez y en la maqueta del software incluida en el Anexo 5, se ha construido a partir de la identificación de las características más importantes (por su efectividad en el estímulo de generación de ideas) del software actualmente disponible, descubiertas por su aplicación en la actividad específica del desarrollo de nuevos productos y siendo utilizado por ingenieros y diseñadores dentro de su propio contexto hispanoamericano.

Se puede enseñar como producto adicional de la investigación, el modelo de evaluación del proceso creativo en la fase conceptual del diseño, presentado en el capítulo siete. Su forma de aplicación y su validez quedaron demostradas por los resultados obtenidos y presentados con detalle en los capítulos ocho y nueve.

La metodología propuesta inicialmente y ajustada durante el desarrollo de la investigación, también puede considerarse como un producto relevante que puede ser aprovechado para estudios similares posteriores.

Se concluye, por lo tanto, que se ha logrado cumplir cabalmente con los objetivos, tanto generales como específicos, propuestos para esta investigación.

## **11.6 Recomendaciones para futuros desarrollos**

La literatura científica muestra un creciente interés por la investigación en diseño, revelado por las múltiples propuestas para abordarla desde diversas perspectivas (psicológico-cognitivas, científico-técnicas, sociales, comerciales). Una muestra de tal interés es el creciente número de grupos y de centros de investigación cuyas actividades giran en torno a tres enfoques: la gestión del conocimiento, la participación de equipos «transdisciplinarios» y la síntesis como etapa determinante del diseño innovador.

La presente investigación se ha orientado por el tercero de aquellos enfoques y se espera que los aportes aquí dados sirvan de estímulo para la continuidad en esta línea de trabajo. Para ello, las recomendaciones específicas sobre temas para futuros desarrollos se pueden concretar de la siguiente manera:

### ***a. Elaboración del software***

A partir de las pautas y de la estructura básica propuesta como resultado en esta investigación, es obvio que la primera recomendación es la elaboración del software. Ello implicará no solo la codificación misma del programa, sino también la construcción de las bases de datos y demás elementos que lo conforman, así como la evaluación del prototipo en ambientes de aplicación real.

### ***b. Búsqueda semántica***

El tema de la automatización de la gestión de información, en particular para la aplicación en el proceso de búsqueda de patentes de invención, catálogos comerciales, etc., se considera de mucha importancia, debido a la actual explosión de información disponible y a la consecuente imposibilidad de revisión y selección por métodos convencionales. Es claro que existe una oferta comercial de software que se aplica a la denominada minería de datos, pero lo que aquí se propone es una investigación sobre alternativas menos costosas que lleven al desarrollo de módulos incorporables al software de diseño.

### ***c. Evaluación de la aplicación de «papel y tinta electrónica» para diseño***

Una de las grandes debilidades que tienen los programas actuales de asistencia a la ingeniería de diseño es la forma no natural que se emplea para interactuar con el

ordenador. Se sugiere llevar a cabo un estudio de la aplicabilidad del nuevo hardware que utiliza la tecnología del papel y de la tinta electrónica como interfaz gráfica para soportar el diseño. Por ello, si el hardware comentado puede llegar a suplir este medio de interacción, la efectividad del software se multiplicará.

***d. Estudio de aplicación del «virtual clayng» en ingeniería de diseño***

La manipulación de objetos es una forma efectiva para transformar ideas vagas o abstractas en soluciones concretas. Muchas veces resulta muy costoso aproximarse a una forma manipulable de un objeto para inducir aquel proceso. Para este propósito podría explorarse la aplicabilidad de esta técnica que está siendo desarrollada en la actualidad.