

HERRAMIENTAS CASE. ¿CÓMO INCORPORARLAS CON ÉXITO EN NUESTRA ORGANIZACIÓN?

Pascual González López

Ana Amelia González López

José Antonio Gallud Lázaro

Pascual González López, Ana Amelia González López y José Antonio Gallud Lázaro están en el Departamento de Informática. Esc. Univ. Politécnica de Albacete. Universidad de Castilla-La Mancha.

RESUMEN

En este artículo se analiza un nuevo modo de desarrollo del software basado en su automatización, las *herramientas CASE*. Este tipo de herramientas tienen como principal objetivo facilitar la obtención de software de mayor calidad, a un menor coste. Junto a una breve exposición sobre lo que entendemos por CASE, nos centraremos en el estudio de las tareas asociadas al proceso de selección e implantación de una herramienta de este tipo. Proceso que, por otra parte, es fundamental para asegurar el éxito en la incorporación de estas herramientas dentro de una organización.

1. ¿QUÉ ES EL CASE?

TODO el mundo conoce el famoso dicho popular de «en casa del herrero cuchara de palo», o la historia del zapatero que estaba tan ocupado que no se percató que sus hijos iban descalzos.

Algo parecido a lo que describen ambos dichos populares es lo que les ha sucedido a los ingenieros del software en el pasado, los cuales, aunque han desarrollado sistemas complejos para automatizar el trabajo de otros, no han llegado a aplicar esas mismas técnicas para intentar automatizar su propio trabajo, el desarrollo de sistemas.

Hoy en día podemos decir que el ingeniero de software ha elaborado su primera cuchara metálica o su primer par de zapatos, en la forma de la «ingeniería de software asistida por ordenador» o más popularmente conocida como «CASE». En la actualidad, y siguiendo con el símil del zapatero, no hay tanta variedad de zapatos como nos gustaría;

a veces son muy rígidos y otros incómodos; no son suficientemente sofisticados para los más exigentes y no siempre van con las prendas que tenemos en nuestros armarios; sin embargo podemos decir que ya constituyen una pieza indispensable del guardarropa del ingeniero y, con el tiempo, se harán más confortables, más adaptables al usuario y tal vez con una apariencia totalmente distinta pero al final seguirán realizando la misma función: ayudar a la concepción y construcción de sistemas software.

Es importante tener en cuenta que al igual que en otras ingenierías, la introducción de herramientas cada vez más potentes no sólo cambia los métodos de diseño o construcción, sino que también amplía el horizonte sobre *qué puede ser construido*. No es posible desarrollar sistemas software altamente complejos utilizando métodos manuales, cuando estos sistemas, al igual que el entorno en el que están inmersos, deberán evolucionar y cambiar rápidamente.

Por tanto en el futuro, para seguir siendo competitivas, las corporaciones dependerán de potentes herramientas que les permitan desarrollar y mantener sus sistemas de información, algo que cada vez es más importante y estratégico para mejorar la gestión, calidad, productividad y en definitiva la competitividad de las organizaciones modernas.

El término C.A.S.E. se estableció durante los años 80 en los Estados Unidos como abreviatura de «*Computer Aided Software Engineering*». Este término se hizo popular, al asociarse con potentes herramientas para el desarrollo de sistemas, con las cuales se abría en un principio una puerta de esperanza a multitud de organizaciones donde el desarrollo y mantenimiento del software se había convertido, como decía Brooks [BRO75], en un «hombre lobo», y se veía a estas herramientas como la «bala de plata» que terminase con él.

Las herramientas CASE proponen una nueva filosofía del concepto de ciclo de vida basado en la automatización, para lo cual proporcionan un conjunto de herramientas bien integradas, que, enmarcadas dentro de una determinada metodología, permiten automatizar las fases del ciclo de vida de un sistema software.

Con esta automatización total, se intenta conseguir una mejora en la productividad y en la calidad del producto final. Para alcanzar estas ventajas se define un conjunto de objetivos:

- Automatizar e integrar las tareas de las distintas etapas del ciclo de vida, junto con la gestión de proyectos software.
- Mejorar la calidad mediante la automatización de la comprobación de errores.
- Automatizar la generación de la documentación.
- Facilitar que se pueda compartir la información entre varios proyectos y la reutilización del software.
- Aportar un entorno de desarrollo interactivo.

- Acercar el desarrollo al usuario facilitando la creación de prototipos.
- Simplificar la labor de mantenimiento.

Esta automatización del proceso de desarrollo lleva consigo algunas variaciones en el ciclo de vida, al permitir la verdadera aplicación de ciclos de vida de desarrollo rápido de sistemas y enfatizar las fases de análisis y diseño, a la vez que se reduce notablemente el tiempo utilizado en el resto de fases (ver figura 1). Los prototipos rápidos e iterativos reemplazan a los métodos manuales de definición de requerimientos facilitando la determinación de las necesidades de los usuarios y sirviendo como especificación «viva» del sistema, siempre actualizada, lo que elimina la necesidad de largos documentos de especificaciones basados en la narrativa.

A continuación vamos a aportar una clasificación de las herramientas CASE que permita caracterizar a las distintas herramientas existentes en el mercado.

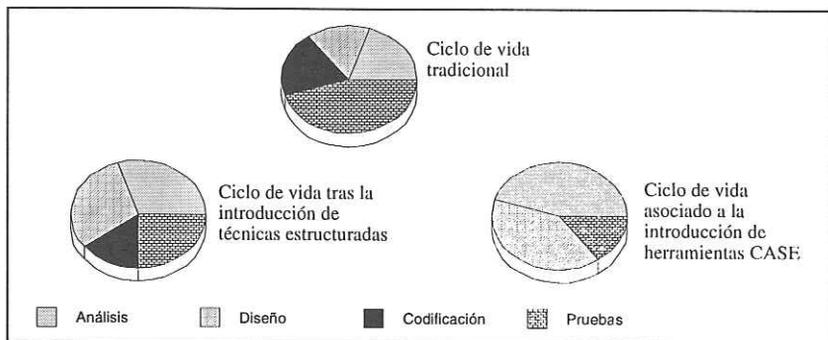


FIGURA 1
Distribución del esfuerzo en las distintas fases según el ciclo de vida

2. CLASIFICACIÓN Y COMPONENTES DE LAS HERRAMIENTAS CASE

Las herramientas CASE se enfocan hacia el soporte de diferentes fases del ciclo de vida del software o al desarrollo de diferentes tipos de sistemas. La disparidad de herramientas que se engloban dentro del concepto CASE hace necesaria una clasificación que nos permita poder realizar una comparación de las distintas herramientas existentes en el mercado. Atendiendo a la clasificación propuesta por Carma McClure [MCC89a] se pueden determinar tres categorías de herramientas:

- **Juego de herramientas (Toolkit):** conjunto de herramientas integradas que automatizan un tipo de tarea del ciclo de vida del software. Constituyen el tipo más simple de herramientas.
- **Banco de trabajo (Workbench):** entorno de propósito general que soporta la totalidad de las tareas del ciclo de vida del software. En este tipo las herramientas se integran de forma que la salida de una fase del ciclo de vida pasa directa y automáticamente a la siguiente.
- **Compañero de metodología (Methodology Companion):** es un tipo de herramientas CASE toolkit o CASE workbench, que estructura el proceso de desarrollo del software de acuerdo con los pasos y reglas de una determinada metodología. Este tipo de herramientas, por ejemplo, informan al desarrollador, mediante menús o pantallas de ayuda del próximo paso a realizar, según la metodología soportada, y no le permiten pasar a él hasta que se compruebe que la tarea actual se ha completado totalmente.

TABLA 1.
Características básicas de una herramienta CASE.

•	Permiten a los usuarios dibujar diagramas para la planificación, análisis, o diseño en la pantalla de un ordenador.
•	Solicitan información acerca de cada uno de los objetos de un diagrama y de las interrelaciones entre dichos objetos.
•	Almacenan el significado del diagrama, en vez del diagrama en sí mismo, dentro de un depósito de información (repository).
•	Comprueban la exactitud, integridad y completitud de cada diagrama. Los diagramas que se ofrezcan deben ser elegidos con el objeto de facilitar esta labor.

Como hemos dicho los juegos de herramientas CASE de tipo toolkits se caracterizan por la realización de algunas tareas específicas dentro del desarrollo del software. Sin embargo, aunque se obtienen beneficios utilizando herramientas aisladas (toolkits) que realicen determinadas labores dentro del desarrollo de sistemas software, la potencia real del CASE sólo puede conseguirse a través de la utilización de las herramientas dentro de un entorno integrado y enmarcado dentro de una metodología de desarrollo, es decir, mediante una herramienta CASE Workbench de tipo Methodology Companion, a lo que más usualmente se le llama **Integrated-CASE (I-CASE)**.

Por tanto la característica esencial de las I-CASE no es otra que la integración, a través de una enciclopedia común, de las distintas herra-

mientas, facilitando, de este modo, que el generador de código se nutra de la información obtenida por las herramientas de diseño, o que se pueda compartir la información de la enciclopedia por diferentes proyectos realizados en paralelo. Entre otras cosas estas características permiten que se consiga una alta productividad, mayor que en aquellas herramientas que no están acopladas.

Una I-CASE, como ya hemos dicho anteriormente, no es una mera reunión de las mejores herramientas disponibles, sino que éstas deben abarcar todas las fases del ciclo de vida. Deben integrarse bajo la cobertura de una interfaz de usuario homogénea y la información debe ser compartida en una enciclopedia común gestionada de forma ordenada, segura y automatizada.

La enciclopedia es el corazón del I-CASE, siendo la que facilita la integridad y consistencia. En ella se almacena, de una manera estructurada, toda la información conforme se va obteniendo en las distintas fases del ciclo de vida. Almacena el significado de los diagramas y comprueba la consistencia de cada representación, eliminando información redundante y permitiendo la reutilización no solo del código sino del análisis o del diseño. Es una utilidad «viva» en la que el conocimiento que se almacena es vital para comprender y modificar los sistemas de una organización, ya que cualquier modificación hace que ésta se actualice automáticamente y que el código sea generado a partir de dichas modificaciones.

La etapa de generación de código tampoco se encuentra aislada en un entorno I-CASE. Los aspectos de mantenimiento del código generado (gestión de versiones y modificaciones realizadas), librerías de componentes y criterios de reutilización, ingeniería inversa y reingeniería de aplicaciones preexistentes, son aspectos de vital importancia en este tipo de entornos.

Pero las capacidades de las herramientas CASE van más allá de una enciclopedia o un generador de código. Las verdaderas potencialidades de estas herramientas pueden verse claramente al estudiar los componentes, o capacidades que pueden ofrecer a los desarrolladores y que, como veremos a continuación, van desde las capacidades gráficas hasta el soporte del mantenimiento de los sistemas actualmente en explotación.

Estas capacidades que proporcionan un soporte total al desarrollo del software pueden resumirse en las siguientes:

- Capacidades gráficas para representar los modelos y diagramas.
- Comprobación de errores integrado dentro de las herramientas.
- Depósito de información o enciclopedia para almacenar y gestionar toda la información del sistema de forma global.

- Conjunto altamente integrado de herramientas para asistir a todas y cada una de las fases del ciclo de vida, dentro de un interfaz común de cara al usuario.
- Soporte de una metodología.
- Generación automática de código desde las especificaciones del diseño y soporte en la obtención de prototipos.
- Soporte de herramientas para el mantenimiento de sistemas antiguos: reestructuración (restructuring), ingeniería inversa (reverse engineering), reingeniería (reengineering).

Aunque sería aconsejable que la herramienta seleccionada contenga los componentes anteriores, la base y por tanto el corazón de la herramienta será la enciclopedia. Las características de ésta serán las que condicionarán las posibilidades de la herramienta y, por tanto, es el componente que con mayor cuidado se debe estudiar en la selección de una herramienta dada.

3. CONSIDERACIONES SOBRE LA ELECCIÓN E IMPLANTACIÓN DE UNA HERRAMIENTA CASE

Antes de abordar de manera detallada el análisis de la elección e implantación de una herramienta CASE cabe preguntarse por la madurez de esta nueva tecnología y repasar algunos de los posicionamientos más comunes ante cualquier nueva tecnología.

El problema de las nuevas tecnologías es que la gente está o muy entusiasmada, y por tanto puede esperar beneficios inalcanzables, o demasiado pesimista y por tanto obstruccionista en su adopción. Hay que tener en cuenta que ambas posturas son ciertamente peligrosas y deben tenerse presentes a la hora de pensar en introducir dentro de una organización una nueva tecnología, en este caso una herramienta CASE.

A la hora de introducir una nueva tecnología en una organización se suelen dar tres modos e instantes de adoptarla ([MAR89], pp. 170-171)

- Los *pioneros* asumen el gran coste inicial de la investigación con el objetivo de obtener una ventaja significativa sobre la competencia. Este coste inicial y sobre todo la incertidumbre de los resultados a obtener hace que sean pocos los que se sumen al tren de la tecnología en sus primeras etapas. En cualquier caso, aunque el riesgo es alto hay que tener en cuenta que si el resultado es satisfactorio los beneficios pueden ser igualmente altos.
- La *segunda ola* se refiere a las corporaciones que introducen esta nueva tecnología cuando los pioneros empiezan a obtener resultados satisfactorios. En ese momento ya hay alguna base

más o menos sólida que aprender y sobre la que trabajar. En este caso los costes y los riesgos se han reducido y sin embargo los beneficios siguen siendo altos, ya que el número de compañías que se incorporan en esta situación es escaso (alrededor del 20%) y por tanto se pueden conseguir considerables ventajas sobre aquellas empresas de la competencia más rezagadas.

- Por último, los *rezagados* piensan en términos de la tecnología actual y no ven claro los beneficios que podrían obtener al abandonar la por una nueva que para ellos, aunque sea por desconocimiento, todavía no es muy segura. En este caso este grupo de organizaciones corren el riesgo de quedarse obsoletas en un mundo tan cambiante. Su incorporación a la nueva tecnología, se realiza generalmente de forma precipitada y desordenada, obligadas por la propia competencia, obteniendo beneficios sensiblemente menores, al haberse quedado rezagadas respecto a las que se incorporaron con anterioridad.

Existen muchos estudios sobre la oportunidad y beneficios obtenidos con la implantación de una herramienta CASE dentro del proceso de desarrollo de sistemas, pero estos estudios no son todos halagüeños y la experiencia nos demuestra que las herramientas CASE no son por sí solas las «*balas de plata*» que desesperadamente buscan algunos directivos.

Por tanto, podríamos preguntarnos ¿qué ha sucedido a la hora de adoptar una determinada herramienta CASE dentro de una organización?

Tal vez como propone Patrick Loy [LOY93], la principal causa de algunos fracasos ha sido que las expectativas eran poco realistas, ya que se buscaba la barita mágica que sin ningún esfuerzo ni compromiso solucionara todos los problemas, y además los productos no estaban preparados para ofrecer lo que los vendedores decían que hacían.

Esta idea general puede resumirse en tres aspectos críticos del problema de las herramientas CASE [LOY93]:

- Hay un sentimiento generalizado de que las máquinas son más fiables que el ser humano y que pueden sustituirlo en ciertas funciones. Este sentimiento puede hacer que los directores de las empresas sean presa fácil de algunos vendedores que prometen la luna.

Pero hay que ser conscientes de que tanto los métodos como las herramientas son de gran ayuda para los ingenieros del software, pero no dejan de ser una ayuda, no pudiendo considerarse como sustitutos de la mente de los desarrolladores.

- Hay una tendencia a pensar que las herramientas reemplazan a los métodos, en vez de, que les sirven a ellos. Esto lleva a muchas organizaciones a pensar que el entrenamiento y soporte

aportado por el vendedor es lo único necesario, lo cual generalmente suele ser un desastre, ya que debería pensarse que el método debe «dirigir» a la herramienta y no a la inversa.

- Las herramientas no soportan todavía de forma adecuada los métodos para los que dicen estar creadas.

A estos tres problemas se puede añadir que muchas organizaciones han adoptado, o al menos comprado, una herramienta CASE motivados por una *moda* y no han tenido en cuenta la complejidad de los problemas del desarrollo del software, lo cual era casi asegurarse el fracaso.

Aun existiendo estos problemas, algunos de los cuales son debidos a una mala comprensión de lo que suponen realmente las herramientas CASE, éstas son una importante ayuda, pero no la solución total y por tanto la panacea, en el proceso de desarrollo. Por tanto, podemos considerar que en la actualidad nos encontramos ante la «*ventana de la oportunidad*». Esto significa que el que sea capaz de incorporarse de forma ordenada y meditada a esta nueva tecnología podrá sacar ventaja a la competencia y evitará los riesgos de incorporarse motivado por una imposición del mercado que le obligue a hacerlo de forma rápida y tal vez incontrolada, con el riesgo adicional que esto implica.

Al igual que sucede con la adopción de una metodología de desarrollo, hace falta un estudio muy serio antes de decidirse por una herramienta CASE en concreto. Hay que ser consciente de qué es lo que está fallando y de cuáles son las necesidades reales y las características particulares de cada organización, algo que en la mayoría de los casos no se plantea nunca de forma seria. Pero esta decisión no resulta simple, pues lleva consigo un importante compromiso, y no debe dejarse llevar por posibles *modas* o tendencias, que mal asimiladas no son sino un estorbo más a la hora de solucionar los problemas del desarrollo de sistemas software, o por *desconfianzas* a ultranza de aquellos que se han sentido engañados anteriormente con otra supuesta panacea.

Entonces cabe preguntarse ¿qué hacer para tomar una decisión lo más acertada posible? En este sentido, una estrategia que puede ayudarnos a tomar una decisión sensata acerca del empleo de la tecnología CASE en un tiempo razonable es:

- Estudiar y determinar cuáles son los problemas reales, las necesidades y las características peculiares de la organización.
- Buscar y seleccionar un conjunto de soluciones que satisfagan, aunque sólo sea en parte, dichos problemas, necesidades y características.
- Establecer y aplicar un plan de implantación que controle y asegure la correcta adopción de dichas soluciones tecnológicas.

En los siguientes apartados vamos a desarrollar brevemente las características principales de cada una de estas etapas.

3.1. Estudio y determinación de necesidades

La mayoría de las organizaciones se encuentran ante una situación de escepticismo sobre la existencia de soluciones mágicas, pero a la vez esperanzadas en encontrar una realmente mágica.

La causa de esta situación de escepticismo es la gran inversión realizada por algunas de ellas y los escasos rendimientos, respecto a lo esperado, de dichas inversiones. A la vez tienen necesidad de soluciones mágicas ya que la rápida evolución de la sociedad, y por tanto de los negocios, les está enterrando entre un montón de solicitudes de nuevos y más complejos sistemas, de mayor calidad, y más baratos. Mientras tanto los centros de desarrollo tienen que hacer frente a las tareas de mantenimiento de los sistemas existentes, tareas que consumen la mayor parte de los recursos.

Ante esta situación seguir esperando la solución mágica puede ser un grave error que sólo puede conducir a que los problemas y los gastos aumenten. Por tanto, aunque es difícil de conseguir, pues el tiempo es un recurso escaso y muy valioso, toda organización que se encuentre en esta situación debería emplear algún tiempo en recapacitar y determinar qué va bien y qué va mal dentro del desarrollo de sistemas software. Este no es un tiempo perdido, sino que debería ser una práctica común para conseguir mejorar y actualizar el proceso de desarrollo de sistemas software dentro de una organización.

Al igual que en otras áreas de una organización, dentro del desarrollo del software cada corporación sufre su propia y personal forma de crisis del software (no hay una solución universal).

Podemos asimilar el problema a lo que sucede en el área de personal o contabilidad de una empresa a la hora de introducir una nueva tecnología, en este caso la mecanización de los distintos procesos asociados a cada una de dichas áreas. Nos encontramos, por tanto, ante la necesidad de mecanizar un conjunto de tareas y para dar solución a esta mecanización existen en el mercado un conjunto de soluciones que en distinta medida podrán cubrir las necesidades de cada área, claro está que antes debemos saber cuáles son estas necesidades. ¿Qué es lo que un conector de sistemas software haría? Pues está claro que el responsable de este estudio intentaría, junto con los responsables del área a estudiar, determinar cuáles son los grandes objetivos de la empresa y dentro de éstos cuáles son las necesidades y peculiaridades de su nomina, contabilidad, etc, para, en función de dichas características, poder seleccionar, dentro de la oferta del mercado, una herramienta que cumpla dichas necesidades con un costo razonable.

Al igual que pediríamos a los responsables del área de personal, que describieran con la mayor precisión posible sus necesidades, del mismo modo debemos pedir a los responsables del área de desarrollo de sistemas software que detallen sus necesidades, ya que decir simplemente que se necesita aumentar la calidad y la productividad no es una forma adecuada para determinar cuál puede ser la solución ideal.

Pero aquí tal vez nos encontramos con un problema añadido, ya que puede no existir dentro de la organización un modo estándar de atacar las tareas de desarrollo de sistemas. Eso es lo mismo que pensar que dentro de una empresa cada empleado del área de personal tuviera una forma propia de obtener la nómina de los empleados. En este punto ¿cómo determinar las necesidades? Si nos encontramos ante una situación tan caótica como la descrita anteriormente, situación más frecuente de lo que puede parecer, cabría pensar que antes de embarcarse en la adopción de una herramienta CASE habría que pensar en normalizar y estandarizar el proceso de desarrollo, bien implantando una metodología estándar, o bien creando nuestra propia metodología. Por tanto en este caso estaríamos ante una pregunta previa que debemos resolver antes de seguir adelante ¿qué metodología, o tal vez metodologías, de desarrollo se debe utilizar de forma estándar en nuestra organización?

Suponiendo que hemos superado esa primera cuestión, es decir, existe una forma estándar de entender el desarrollo de sistemas, nos encontramos en el mismo punto que están los responsables del área de personal que pretenden introducir dentro de su entorno de trabajo una nueva tecnología.

Se deben estudiar tanto las necesidades de tipo organizativo como técnico, para de este modo tener una comprensión mayor del problema y poder adoptar una solución más coherente y adecuada. Para determinar dichas necesidades podemos efectuar un conjunto de preguntas fuertemente interrelacionadas, entre las que podemos destacar:

- ¿Cuál es el método estándar de trabajo?, ¿este método se utiliza siempre en todos los desarrollos de sistemas software?, ¿existen controles que aseguren la correcta utilización de dicho método?
- ¿Qué problemas encuentran los desarrolladores en la aplicación del método?
- ¿Cuáles son las quejas más frecuentes de los usuarios respecto a los sistemas desarrollados?
- ¿Qué sistemas funcionan correctamente y cuáles son las características (en cuanto a desarrollo, organización, mantenimiento, etc) que pueden diferenciarlos de aquellos que no funcionan como se desea?
- ¿Existen formas de medir la productividad de los sistemas desarrollados?, si es así ¿cuál es su productividad?

- ¿Qué estándares y procedimientos técnicos existen para garantizar la calidad del software?
- ¿Cuál es la formación y predisposición del personal de desarrollo ante la inclusión de nuevas tecnologías (cambios)?
- ¿Qué herramientas de desarrollo y mantenimiento del software, incluyendo los lenguajes de programación, se están utilizando?, ¿cuál es la satisfacción en cuanto a su funcionamiento?, ¿se pretende que continúen siendo utilizadas o pueden ser sustituidas?
- ¿Cuál es la estructura organizacional de los desarrollos software?, ¿es centralizada, descentralizada o mixta (centraliza los desarrollos corporativos y descentraliza desarrollo puntuales de un área o departamento)?
- ¿Cuál es la predisposición de la organización a introducir y por tanto a invertir en nuevas tecnologías?

Estas y otras preguntas deberían hacerse los responsables de los departamentos de desarrollo de sistemas software antes de decidir si adoptar o no una herramienta CASE y en su caso cuales son sus requerimientos.

3.2. Búsqueda y selección de una herramienta CASE

Una vez que se conocen las necesidades y características propias del departamento de desarrollo y se ha decidido adoptar una herramienta CASE, el siguiente paso es buscar dentro del mercado cuáles son las candidatas que mejor se ajustan a nuestras necesidades.

En este punto es necesario resaltar que la característica que más nos va a condicionar a la hora de seleccionar una herramienta y que por tanto va a restringir el ámbito de búsqueda, es la metodología que se haya adoptado como estándar dentro del desarrollo de sistemas. Esto es esencial pues generalmente una herramienta CASE está pensada para dar soporte a un conjunto de técnicas propias de una metodología e incluso puede incluir dentro de sus características la de servir de ayuda en la aplicación de una determinada metodología (methodology companion).

Otra característica casi general es la dificultad de encontrar una herramienta que se adapte completamente a nuestras necesidades, aún en el caso de que la metodología de desarrollo aplicada dentro de la organización sea una estándar. En este sentido hay que tener en cuenta cuáles son las restricciones que el uso de la herramienta supondrá en la aplicación completa de la metodología.

A la vez que consideramos los puntos anteriores deben tenerse en cuenta otras características comunes a la selección de cualquier herramienta mecanizada en cualquier entorno, como por ejemplo: si es fácil

de usar y aprender, cual es el soporte técnico y de aprendizaje que se ofrece, cómo se adapta a las necesidades planteadas, si contempla al menos las consideradas mínimas, cuál es el ratio coste/beneficio, etc.

Una vez que se han tenido en cuenta estas u otras cuestiones los responsables pueden determinar cuál es la herramienta que más se adapta a su organización.

3.3. Establecimiento y aplicación de un plan de implantación

Una vez elegida una determinada herramienta no está todo hecho, sino que es ahora cuando debe tenerse más cuidado para asegurar el correcto funcionamiento e integración de la herramienta dentro de la organización. Para ello hay que establecer un plan que controle y garantice una correcta implantación. Este plan debe establecer un conjunto de fases que faciliten la incorporación paulatina de este *elemento extraño* dentro de la organización.

Como una implantación global, a todo el personal, es algo muy difícil de conseguir, lo primero que deberemos hacer es seleccionar un grupo de personas que constituirá el embrión desde donde se extenderá esta nueva tecnología al resto de la organización. La correcta selección del personal que constituirá este grupo es algo importante pues de ellos dependerá en gran medida el éxito de la implantación de la herramienta. Debemos seleccionar a personas que conozcan bastante bien la metodología de trabajo y tengan una predisposición favorable a la incorporación de este tipo de herramientas de ayuda.

Una vez seleccionado el grupo de personas que constituirán la avanzadilla de la implantación lo siguiente a tener en cuenta es su formación, para lo cual será necesario planificar un conjunto de cursos de aprendizaje que les permitan familiarizarse con la herramienta.

Tras esta primera toma de contacto con la herramienta es interesante que este grupo de personas lleven a cabo el desarrollo de un proyecto piloto. Éste debe ser un proyecto real, no crítico, elegido con el objetivo de probar la herramienta, o las herramientas si se han seleccionado más de una, y por tanto de un tamaño apropiado para conseguir en un tiempo mínimo una opinión sobre cómo se puede adaptar una herramienta a las necesidades de la organización.

Una vez terminado el o los proyectos pilotos los resultados de estos servirán para:

- Sugerir cuál es la herramienta, en el caso de haber seleccionado más de una, que más se adapta y cómo a nuestras necesidades.
- Convencer a la dirección y al personal de desarrollo más reacio a su implantación, de las bondades y beneficios que pueden derivarse de su adopción.

Si tras la presentación de los resultados obtenidos, la decisión de la dirección es continuar con su implantación deberían realizarse un conjunto de acciones que asegurasen su correcta expansión e integración:

- Definir un conjunto de procedimientos estándar que puedan utilizarse para la aplicación de esta nueva tecnología.
- Proponer un plan de formación y de nuevos proyectos a abordar que garantice la implantación general de la herramienta seleccionada dentro de la organización.

Dentro de esta etapa de expansión y consolidación de la herramienta sigue jugando un papel esencial el grupo inicial de implantación, ya que éste deberá integrarse en los nuevos proyectos y ser el soporte básico a las consultas y posibles problemas que surjan en el desarrollo de estos nuevos proyectos.

Al mismo tiempo este plan deberá incorporar un conjunto de controles y revisiones que garanticen la correcta utilización de las herramientas y metodologías dentro del entorno de desarrollo y que detecten posibles deficiencias que surjan durante su aplicación y, por tanto, puedan servir para la continua y necesaria actualización y mejora del proceso de desarrollo.

4. CONCLUSIONES

Como hemos visto, las herramientas CASE como toda nueva tecnología crea ciertas actitudes que pueden dificultar su correcta difusión. En cualquier caso, aunque la madurez de esta tecnología es suficiente para pensar en su introducción dentro del proceso de desarrollo, no debemos olvidar que para garantizar su éxito no podemos introducirla de un modo desordenado, sino que su incorporación debe ir acompañada de una estrategia de selección e implantación correcta.

Este proceso de selección e implantación de una herramienta CASE no difiere en gran medida del proceso seguido para elegir y poner en funcionamiento otro paquete software cualquiera. Tal vez la diferencia estriba en la naturaleza y complejidad de la herramienta y en las dificultades derivadas de la homogeneización del trabajo en el área de desarrollo de software. En todo caso, esa dificultad no debe ser un obstáculo, ya que la introducción de este tipo de herramientas no solo ofrecen la posibilidad de automatizar el proceso de desarrollo, sino que también nos ofrecen la posibilidad de estudiar detenidamente dicho proceso dentro de nuestra organización. Este estudio profundo del proceso de desarrollo sirve realmente para conseguir una estandarización del mismo y la eliminación de algunos de los problemas que más frecuentemente aparecen en la mayoría de los desarrollos. De este modo a las ventajas asociadas a la automatización se pueden incorporar las de-

rivadas de la realización de un estudio profundo del propio método de desarrollo.

BIBLIOGRAFÍA

- [BRO75] BROOKS, F.: *The Mythical Man-Month*. Addison-Wesley, 1975.
- [GAN90] GANE, C.: *Computer-Aided Software Engineering. The methodologies, the products, and the future*. Prentice-Hall International, 1990.
- [GIB89] GIBSON, M. L.: «La filosofía CASE». *Binary*, nº 10, pp. 82-88.
- [LOY93] LOY, P.: «The Method Won't Save You (but it can help)». *ACM SIGSOFT*, vol. 18, nº 1, Jan 1993, pp. 30-34.
- [MAR89] MARTIN, J.: *Information Engineering. Book I Introduction*. Englewood Cliffs, Prentice-Hall, 1989.
- [MAR90b] MARTIN, J.: *Information Engineering. Book III Design & Construction*. Englewood Cliffs, Prentice-Hall, 1990.
- [MCC89a] McCLURE, C.: *CASE is Software Automation*. Englewood Cliffs, Prentice-Hall, 1989.
- [MCC89b] McCLURE, C.: «La experiencia CASE». *Binary*, nº 10, pp. 103-111.
- [PRE93] PRESSMAN, R.: *Ingeniería del Software. Un enfoque práctico*. McGraw-Hill, Madrid, 1993.
- [SAL88] SALINAS, A.: «Cómo elegir e implantar herramientas CASE de automatización del software». *Computerworld*, 26, Febrero, 1988, pp. 21-23.
- [SLA93] SLAVKOVA, O., GRANE, J.: «To CASE, or not to CASE?». *Novática*, nº 105, Sept./Oct., 1993, pp. 3-9.
- [SUL91] SULLIVAN-TRAINOR, M.L.: «Buyers' Scorecard. TI's IEF scores high for integration, benefits delivery». *Computerword*, April 22, 1991, pp. 72-74.