



HERRAMIENTAS CASE

1. Introducción
2. Categorías de herramientas CASE
3. Integración de herramientas CASE
4. Implantación de CASE
 - 4.1. Causa del fracaso en la adopción CASE

Bibliografía: Piattini, M., Calvo-Manzano, J., Cervera, J. y Fernández, L. (2004). *Análisis y Diseño de Aplicaciones Informáticas de Gestión: Una perspectiva de Ingeniería del Software*. Madrid: Ra-Ma.



HERRAMIENTAS CASE

1. INTRODUCCIÓN

Desde que a finales de los años sesenta se acuña el término “crisis del software”, numerosos expertos han estado ocupándose del tema, proponiendo distintas técnicas, metodologías y herramientas para paliar esta situación.

Entre todas ellas destaca la tecnología conocida con el nombre de **CASE** (*Computer Aided/Assisted Software/System Engineering*) que en su acepción más amplia se puede definir como:

“ el conjunto de herramientas y metodologías que soportan un enfoque de ingeniería para las distintas fases del desarrollo de software”



HERRAMIENTAS CASE

- Esta tecnología surge a mediados de los años setenta, cuando empiezan a aparecer las primeras metodologías estructuradas y se inician las investigaciones sobre entornos de desarrollo.
- A mediados de los años ochenta, se populariza y surgen las primeras herramientas de documentación y diagramación automática. Es una época en la que explotan el número de seminarios, cursos, revistas, libros y congresos dedicados al tema.
- También ha jugado un papel importante en este desarrollo la aparición de las estaciones de trabajo, que aportan un buen interfaz gráfico asociado a una gran capacidad de proceso, dos de los requisitos básicos para el CASE.

Profesora: Marcela Genero Bocco

3



HERRAMIENTAS CASE

- En los ochenta surge asimismo el concepto de repositorio como núcleo de un entorno CASE, así como generadores de programas y aplicaciones que automatizan gran parte de las últimas fases del ciclo de vida. En paralelo también aparecen los gestores de proyectos, algunos de los cuales se integran con herramientas CASE.
- A finales de esa década se produce un considerable aumento en la venta de estos productos y empieza la etapa de asimilación de la tecnología, que fracasa debido a las limitaciones de la “primera” generación de productos.

Profesora: Marcela Genero Bocco

4



HERRAMIENTAS CASE

Esta *primera generación* de herramientas fracasa, debido principalmente a tres factores:

- limitaciones de los productos
- falsas expectativas sobre sus posibilidades
- incorrecta implantación

A estos factores es preciso añadir la resistencia al cambio que habitualmente se produce cuando aparece un nuevo paradigma que obliga a realizar un importante esfuerzo de formación y, especialmente, en adaptación de la mentalidad de los usuarios.



HERRAMIENTAS CASE

En definitiva, la tecnología CASE ha experimentado la clásica evolución que sufren aquellos paradigmas (como técnicas estructuradas, inteligencia artificial, lenguajes de cuarta generación y, en estos momentos, la orientación al objeto), que se ofrecen como la panacea universal capaz de resolver todos los problemas del desarrollo de sistemas de información.

A mediados de los noventa esta tecnología entró en su fase de madurez en la que surgió una “*segunda generación*” de herramientas (algunas de las cuales ya no aparecen bajo el término CASE, para no rememorar el fracaso anterior), que superan gran parte de las limitaciones existentes en la primera generación.



HERRAMIENTAS CASE

Además, los usuarios conocen mejor sus posibilidades y han aprendido a poner unas expectativas más justas sobre éstas, mejorando también los procesos de adopción de metodologías y herramientas.

La tecnología CASE supone la "*informatización de la informática*", es decir "*la automatización del desarrollo del software*", contribuyendo así a elevar la productividad y la calidad en el desarrollo de sistemas de información, de forma análoga a lo que suponen las técnicas CAD/CAM en el área de fabricación.

- CAD (Computer Aided Design), diseño asistido por ordenador.
- CAM (Computer Aided Manufacturing), fabricación asistida por ordenador.

Profesora: Marcela Genero Bocco

7



HERRAMIENTAS CASE

Este nuevo enfoque a la hora de construir software, persigue los siguientes objetivos:

- permitir la aplicación práctica de metodologías estructuradas, lo que resulta muy difícil sin emplear herramientas
- mejorar la calidad del software
- facilitar la realización de prototipos, y el desarrollo conjunto de aplicaciones
- simplificar el mantenimiento de los programas
- estandarizar la documentación
- aumentar la portabilidad de las aplicaciones
- facilitar la reutilización de componentes software
- permitir un desarrollo *visual* de las aplicaciones, mediante la utilización de gráficos

Profesora: Marcela Genero Bocco

8



HERRAMIENTAS CASE

De una manera muy esquemática, se puede afirmar que una herramienta CASE se compone de los siguientes elementos:

- **Repositorio** (diccionario) donde se almacenan los elementos definidos o creados por la herramienta, y que se basa en un SGBD o en un sistema de gestión de ficheros.
- **Metamodelo** (no siempre visible), que define las técnicas y metodologías soportadas por la herramienta, y que es conveniente que pueda ser extensible por parte del usuario
- **Generador de informes**, que permite obtener toda la documentación que describe el sistema de información desarrollado; documentación que está asociada a las técnicas y metodologías

Profesora: Marcela Genero Bocco

9



HERRAMIENTAS CASE

Herramienta de carga/descarga de datos, que permite cargar el repositorio de la herramienta CASE con datos provenientes de otros sistemas, o generar a partir de la propia herramienta esquemas de bases de datos, programas, etc.

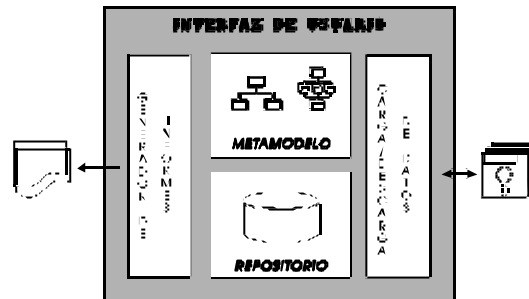
- **Interfaz de usuario**, que constará de editores de texto y herramientas de diseño gráficos, que permitan mediante la utilización de un sistema de ventanas, iconos y menús, con la ayuda del ratón, definir los diagramas, matrices, etc. que incluyen las distintas metodologías. Lo que se conoce usualmente por las siglas inglesas WIMP (Windows, Icons, Mouse y Pull-down menus).
- **Comprobación de errores**, facilidades que permiten llevar a cabo un análisis de la exactitud, integridad y consistencia de los esquemas generados por la herramienta.

Profesora: Marcela Genero Bocco

10



HERRAMIENTAS CASE



Componentes de una herramienta CASE

Profesora: Marcela Genero Bocco

11



HERRAMIENTAS CASE

2. CATEGORÍAS DE HERRAMIENTAS CASE

Como sucede en otras áreas de la informática, la tecnología CASE emplea una terminología que puede resultar a veces confusa, por la que existen numerosas clasificaciones de las herramientas CASE, aunque se pueden distinguir, en general, tres categorías de herramientas:

- Herramientas de gestión, encargadas de la estimación y planificación del proyecto.
- Herramientas técnicas, que se dividen tradicionalmente en:
 - **CASE frontales** ("front-end") o **superiores** ("Upper CASE"), abarcan las primeras fases de análisis y diseño
 - **CASE dorsales** ("back-end") o **inferiores** ("Lower CASE"), cuyo objetivo suele ser el diseño detallado y la generación de código

Profesora: Marcela Genero Bocco

12



HERRAMIENTAS CASE

➤ Herramientas de soporte, como el sistema de repositorio/diccionario, control y configuración, seguridad, etc.

Se denomina **ICASE** (*Integrated CASE*) a las herramientas que engloban ambos aspectos, e **IPSE** (*Integrated Programming Support Environment*) a aquellas que, además, incluyen componentes para la gestión de proyectos y la gestión de configuración.

Es importante destacar que las fronteras entre las distintas categorías no son tan nítidas, por lo que un mismo producto puede catalogarse en distintos apartados.

Profesora: Marcela Genero Bocco

13



HERRAMIENTAS CASE

A continuación presentamos de manera muy resumida las categorías de herramientas CASE más frecuentes.

A) Herramientas de análisis y diseño

Las herramientas de análisis y diseño suponen la parte más importante de las herramientas CASE en cuanto a número de licencias vendidas y de fabricantes, y su principal objetivo es ayudar a la definición de requisitos del sistema y sus propiedades.

Dentro de esta categoría destacan las herramientas que permiten crear y modificar diagramas E/R, diagramas de flujo de datos, diagramas de estructura de cuadros, diagramas de clases, etc.

Profesora: Marcela Genero Bocco

14



HERRAMIENTAS CASE

También son muy importantes las herramientas de prototipado como los diseñadores de pantallas, generadores de menús, generadores de informes y lenguajes de especificación ejecutables.

Un aspecto a destacar es la capacidad de análisis y verificación de especificaciones que soporta la herramienta, no sólo sintáctica sino también semántica, como, por ejemplo, la capacidad de normalizar un diagrama de datos (usualmente hasta tercera forma normal).

Todas estas herramientas han de operar, sobre un repositorio donde se va almacenando la información necesaria para el funcionamiento de la misma herramienta, y cuyo contenido –más o menos completo-, posibilidad de extensión, capacidades de gestión, etc. Dependerá de cada producto.

Profesora: Marcela Genero Bocco

15



HERRAMIENTAS CASE

B) Generación de código y documentación

A partir de las especificaciones del diseño se puede generar código tanto para los programas (por ejemplo, en C o COBOL) como los esquemas de bases de datos (sentencias de definición en SQL) convenientes. Actualmente, las herramientas CASE ofrecen interfaces con diversos lenguajes de cuarta generación para la construcción de sistemas de manera rápida.

Las herramientas CASE también soportan la creación automatizada de un conjunto muy variado de documentación (obtenido a partir de la información almacenada en el directorio) que va desde la descripción textual de un pseudocódigo hasta diagramas más o menos complejos.

Profesora: Marcela Genero Bocco

16



HERRAMIENTAS CASE

C) Herramientas de prueba

Las herramientas de prueba se conocen también por las siglas CAST (Computer Aided Software Testing), y es un área bastante reciente dentro de la tecnología CASE.

En Poston y Sexton (1992) se presentan algunas funcionalidades que suelen tener este tipo de herramientas:

➤ Gestión de pruebas:

- Predecir coste y tiempo de las pruebas
- Planificar pruebas de productos, actividades y recursos
- Monitorizar pruebas
- Generar informes

Profesora: Marcela Genero Bocco

17



HERRAMIENTAS CASE

➤ Definir requisitos y objetivos de prueba:

- Registrar requisitos y/u objetivos
- Verificar requisitos

➤ Diseñar pruebas:

- Diseñar, generar y documentar los casos de prueba
- “Rastrear” (tracing) requisitos
- Trazar caminos, ramas y sentencias de casos de prueba
- Predecir la salida esperada

➤ Construir entornos de ejecución de pruebas:

- Verificar código
- Implementar código
- Generar elementos de prueba: módulos impulsores (drivers), módulos ficticios (stubs), interfaces, etc.

➤ Ejecutar pruebas:

- Fallos de las pruebas
- Estadísticas de fallos
- Calidad de las pruebas
- Calidad del software (funcionalidad, rendimiento, etc.)

Profesora: Marcela Genero Bocco

18



HERRAMIENTAS CASE

D) Herramientas de gestión de configuración

En entornos de desarrollo complejos, especialmente si se integran diversas herramientas de ingeniería de software, se hace imprescindible la incorporación de una herramienta capaz de gestionar la configuración de los sistemas.

Este tipo de herramientas ofrece cuatro tipos de capacidades:

- Control de versiones, es decir, la capacidad de proporcionar almacenamiento y acceso controlado a los datos, así como de registrar los cambios sobre los mismos y poder recuperar versiones anteriores
- Construcción (make), que consiste en automatizar las tareas asociadas a la compilación y enlace de un sistema con el fin de generar ejecutables.
- Trazabilidad de requisitos y análisis de impacto, que permiten respectivamente, rastrear un requisito hasta su implementación, y conocer los elementos del sistema que se ven afectados ante un cambio.

Profesora: Marcela Genero Bocco

19



HERRAMIENTAS CASE

E) Herramientas de ingeniería inversa

Dentro de este apartado destacan diversas herramientas, como las que llevan a cabo:

- **Ingeniería inversa de datos**, que son capaces de extraer la información del código fuente que describe la estructura de los elementos de datos; construyendo así diagramas E/R partiendo de esquemas relacionales, en red o, incluso, ficheros.
- **Ingeniería inversa de procesos**, que incluso permiten aislar la lógica de las entidades, y las reglas del negocio a partir del código.

Profesora: Marcela Genero Bocco

20



HERRAMIENTAS CASE

- **Reestructuración de código fuente**, que modifican su formato o implantan un formato estándar.
- **Redocumentación**, que permiten generar diagramas a fin de que se comprenda mejor el código.
- **Análisis de código**, cuyas funcionalidades van desde la identificación automática del código fuente hasta la posibilidad de ir visualizando dinámicamente las llamadas del mismo. –

Profesora: Marcela Genero Bocco

21



HERRAMIENTAS CASE

3. INTEGRACIÓN DE HERRAMIENTAS CASE

- Las herramientas CAASE se pueden utilizar en distintas formas:
- Como herramientas aisladas, en cuyo caso sólo debe abordarse la compatibilidad de los elementos del entorno.
- En pequeños grupos, que se comunican directamente, para los que la integración está predefinida de manera propietaria.
- En presencia de un marco de integración más amplio, en cuyo caso habrá que abordar la capacidad de la herramienta para utilizar servicios relevantes del marco de integración.
- Suele ser habitual tener que enfrentarse a este último caso, debido a que un solo fabricante no suele ofrecer las mejores herramientas para las distintas tareas a realizar durante todo el ciclo de vida, por lo que es necesario integrar diversos productos

Profesora: Marcela Genero Bocco

22



HERRAMIENTAS CASE

Tradicionalmente se distinguen tres clases de integración (Garbajosa y Bonilla, 1995):

- Integración de datos, que establece la posibilidad de que una herramienta utilice o entregue información a otras herramientas, bien sea mediante un mecanismo de transferencia de datos o a través de un repositorio.
- Integración de control, situación en la que una herramienta puede invocar a otras y comunicarles eventos.
- Integración de presentación, que se refiere a la homogeneidad y consistencia de la interfaz de usuario, por ejemplo, que en todas las herramientas se acceda a la ayuda de la misma manera evitando que el usuario se encuentre con sorpresas.

Profesora: Marcela Genero Bocco

23



HERRAMIENTAS CASE

4. IMPLANTACIÓN DE CASE

- La tecnología CASE está experimentando un gran avance en distintas áreas, solventando bastantes de los defectos que presentaba la primera generación de productos. Sin embargo, el principal desafío de esta tecnología sigue siendo su adopción por parte de los profesionales y las empresas.
- Se calcula que en la práctica se abandona la tecnología CASE en un alto porcentaje: el 70% de las herramientas y técnicas se deja de utilizar un año después de su introducción, el 25% lo emplea sólo un grupo o una persona dentro de la empresa, mientras que el 5% restante sí se emplea ampliamente, pero no al 100% de su capacidad.

Profesora: Marcela Genero Bocco

24



HERRAMIENTAS CASE

- No creemos que exista un método infalible para evitar esta situación y asegurar el éxito en la implantación de CASE; de lo que no cabe duda es de que una buena planificación, una cuidada gestión de expectativas, una sólida formación, una considerable inversión (tanto en los productos CASE como en la infraestructura hardware/software necesaria), junto a la participación de los directivos, aumentan la posibilidad de conseguir un resultado de aceptable que de todas formas no ha de plantarse a corto plazo.

4.1. Causas del fracaso en la adopción CASE

a) Deficiencia de la propia tecnología

Como ya hemos señalado, un gran número de empresas que empezaron a utilizar herramientas CASE en los años ochenta, posteriormente las abandonaron debido a sus inconvenientes, entre los cuales podemos destacar:

Profesora: Marcela Genero Bocco

25



HERRAMIENTAS CASE

- Soporte parcial del ciclo de vida, lo que permite automatizar sólo parte de las actividades de desarrollo, mientras que otras se siguen realizando de forma tradicional.
- Incompatibilidad entre herramientas, incluso entre distintas versiones de la misma herramienta que no siempre se encuentran “sincronizadas” en todas las plataformas hardware/software sobre las que actúan.
- Escasa integración entre las mismas herramientas, así como entre éstas y el resto del entorno: SGBD, lenguajes de cuarta generación, generadores de informes, etc.
- Poca confianza en el vendedor/distribuidor, ya que algunas empresas de CASE son relativamente pequeñas y corren el peligro de desaparecer o ser absorbidas.

Profesora: Marcela Genero Bocco

26



HERRAMIENTAS CASE

- Escasa e inadecuada documentación generada por la herramienta.
- Gran abundancia de herramientas, señalada muchas veces como inconveniente, ya que produce una especie de “bloqueo” a la hora de comprar la herramienta.
- Funcionamiento deficiente en entornos multiusuarios, ya que muchas herramientas surgieron para ordenadores personales.
- Poca capacidad de adaptación a los usuarios (“customización”).
- Un alto coste, no sólo de la herramienta sino también de la plataforma que exige.

Todas estas deficiencias pueden, actualmente, ser superadas en mayor o menor medida, evaluando varias herramientas, considerando, si fuera posible, un cambio en el hardware/software utilizado intentando cuantificar el coste de la “no adopción” (con especial énfasis en el mantenimiento) y valorando los beneficios que el CASE puede soportar.

Profesora: Marcela Genero Bocco

27



HERRAMIENTAS CASE

b) Deficiencias en la aplicación de la tecnología a los problemas (incorrecta implantación)

- Otra causa de fiasco se debe a la aplicación de herramientas CASE a problemas para los que no están preparadas debido a que:
- Soportan una sola metodología por ejemplo., especializada en el desarrollo de aplicaciones de gestión, y que se pretende emplear para construir sistemas de tiempo real.
- No soportan la técnica más adecuada, por ejemplo en el diseño de bases de datos muy grandes puede ser conveniente emplear la “integración de vistas”, que muchas herramientas CASE no soportan
- Metodologías y herramientas que funcionan relativamente bien en proyectos pequeños o medianos, pueden fracasar en proyectos grandes y viceversa.

Profesora: Marcela Genero Bocco

28



HERRAMIENTAS CASE

➤ La selección se centra sólo en factores técnicos por lo que la herramienta resulta insuficiente para los aspectos relativos a la gestión que todo desarrollo lleva consigo.

Las medidas más eficaces para afrontar estos problemas pueden ser: comprender y analizar los distintos tipos de metodologías y herramientas existentes (junto a su “escalabilidad”), utilizando las herramientas adecuadas a cada problema, lo que supone un gran esfuerzo en formación e inversión en consultoría.

c) Deficiencias de la propia organización (falsas expectativas)

Independientemente de las deficiencias citadas anteriormente que son propias de las herramientas en sí mismas, muchas veces los fracasos en la adopción de herramientas CASE son debidos a problemas de la propia organización. En definitiva, la adopción de la “filosofía” CASE es, como la transferencia de cualquier otra tecnología, un tema más “cultural” que tecnológico.

Profesora: Marcela Genero Bocco

29



HERRAMIENTAS CASE

Las causas de fracaso más notables en esta área:

- Actitud por parte de los directivos, que pretenden introducir la tecnología CASE como la “panacea” o “salvación” de todos los males del desarrollo sin contar con una base metodológica.
- Infravaloración del esfuerzo requerido (económico, formación, aceptación).
- Incapacidad para encontrar las metodologías y herramientas adecuadas del nivel de madurez de la organización.
- Inadecuada formación
- No medir la productividad ni la rentabilidad que resulten de la aplicación de la tecnología.

Estas deficiencias se pueden superar con una gestión adecuada de las expectativas, siendo realista (conociendo la cultura de la empresa y su historia frente a los cambios tecnológicos) y con una buena gestión.

Profesora: Marcela Genero Bocco

30