

Mantenimiento del Software

S1

Francisco Ruiz, Macario Polo

Grupo Alarcos

Dep. de Informática

ESCUELA SUPERIOR DE INFORMÁTICA
UNIVERSIDAD DE CASTILLA-LA MANCHA



<http://alarcos.inf-cr.uclm.es/doc/mso/>

Ciudad Real, 2000/2001



Índice - Sesión 1

- Introducción.
 - Ingeniería, Crisis y Mantenimiento del Software
 - El ciclo de vida del software
 - Concepto de Mantenimiento del Software
- Costes y Causas del MS
 - El efecto Iceberg: costes intangibles
 - Causas del alto coste del MS
- Tipos de Mantenimiento
 - Mantenimiento Correctivo
 - Mantenimiento Adaptativo
 - Mantenimiento Perfectivo
 - Mantenimiento Preventivo
- Actividades del MS
 - Actividades según el Tipo de Mantenimiento

Ingeniería, Crisis y Mantenimiento del Software

- Frente a la considerable velocidad con que se ha desarrollado la ingeniería de computadores (hardware), el desarrollo del software ha sufrido un retraso histórico en cuanto a la elaboración y disposición de un cuerpo de doctrina tecnológico (metodologías y herramientas) y científico (modelos o teorías en los que basar lo anterior).
- En 1970 ya se había popularizado el término **Crisis del Software** para referir esta situación. Los síntomas de esta crisis han estado repercutiendo desde entonces en la industria de desarrollo de software y todavía se sienten sus efectos. Para resolver el problema surgió un área de la informática que recibe el nombre de Ingeniería del Software.
- Una de las principales causas de esta situación ha sido la poca importancia que se le ha dado al proceso de Mantenimiento del Software desde todos los colectivos afectados (gestores de empresas, responsables de centros de proceso de datos, informáticos y usuarios).

La ingeniería del software

- Entre las numerosas definiciones de ingeniería del software existentes en la bibliografía es interesante la formulada por McDermid [1991]: *"Ingeniería del software es la ciencia y arte de especificar, diseñar, llevar a cabo y desarrollar -con economía, prontitud y elegancia- programas, documentación y procedimientos operativos mediante los cuales los computadores pueden ser útiles para el ser humano"*.
- En esta definición llama la atención la inclusión de los aspectos artísticos (creatividad) y económicos. También refleja claramente que la ingeniería del software abarca la obtención de "productos" adicionales al código de los programas.

El ciclo de vida del software

- La complejidad del proceso de producción de software se intenta abordar mediante la descomposición en diversas etapas. Esta descomposición ha recibido el nombre de Ciclo de Vida del Software. Los diversos modelos de ciclo de vida que han sido propuestos plantean variantes a partir de las siguientes fases principales:
 - Análisis y Definición de Requisitos.
 - Especificación.
 - Diseño.
 - Programación (escritura del código).
 - Prueba e instalación.
 - Operación y mantenimiento.
- Por tanto, las tareas de mantenimiento son las últimas en realizarse en el ciclo de vida clásico del software.

Concepto de Mantenimiento del Software

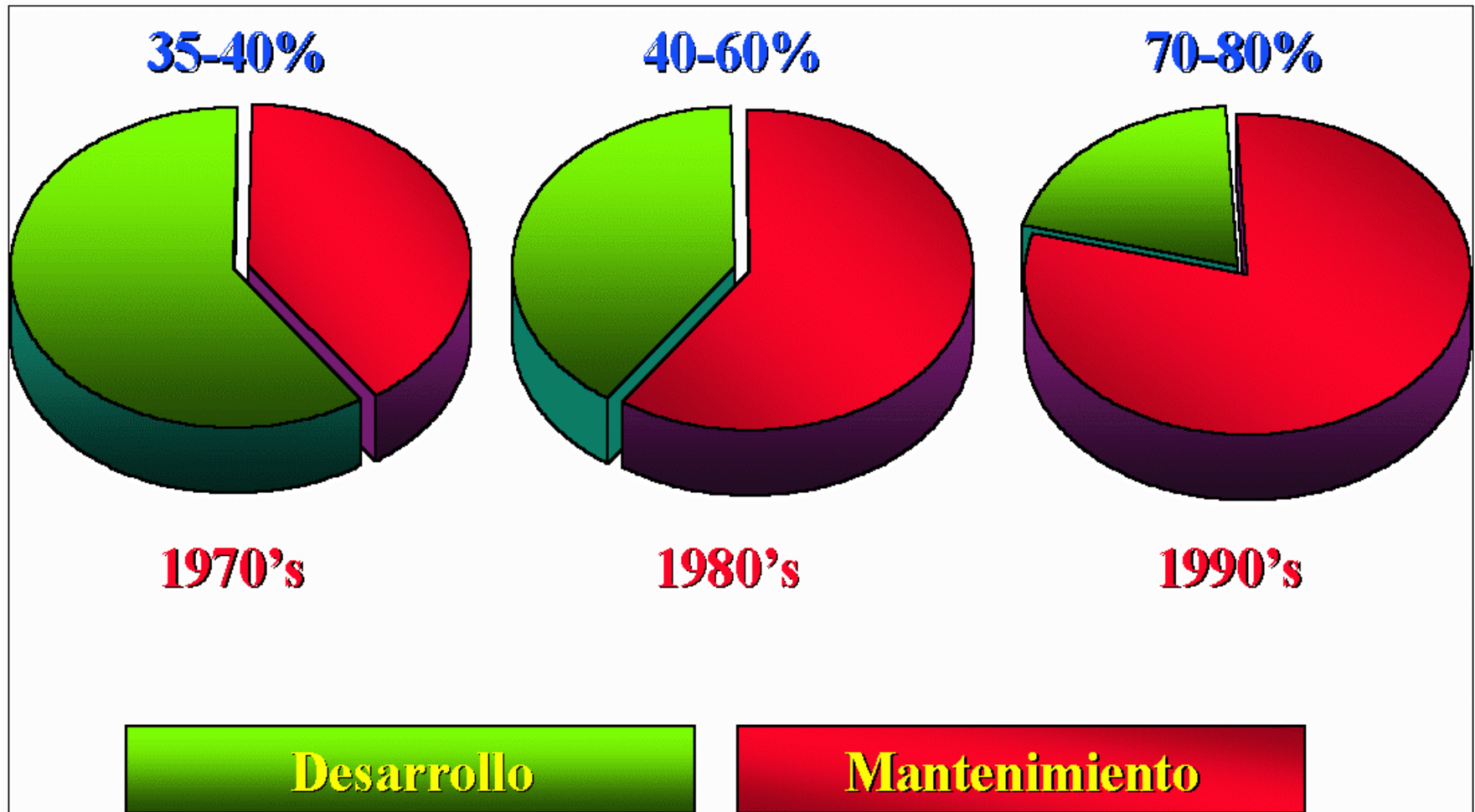
- Aun cuando son las últimas en el ciclo de vida del software, las actividades de mantenimiento no son las menos importantes. Muy al contrario, a continuación veremos que el mantenimiento del software se ha convertido en la principal actividad en cuanto a recursos necesarios y costes.
- Según la terminología ANSI-IEEE, el mantenimiento del software es: *“la modificación de un producto software después de su entrega al cliente o usuario para corregir defectos, para mejorar el rendimiento u otras propiedades deseables, o para adaptarlo a un cambio de entorno”*.

Costes del MS (i)

- Múltiples estudios señalan que el mantenimiento es la parte más costosa del ciclo de vida del software. Estadísticamente está comprobado que el coste de mantenimiento de un producto software a lo largo de toda su vida útil supone más del doble que los costes de su desarrollo. La tendencia es creciente con el paso del tiempo:

<u>Referencia</u>	<u>Fechas</u>	<u>% Mantenimiento</u>
[Pressman, 1993]	años 70	35%-40%
[Lientz y Swanson, 1980]	1976	60%
[Pigoski, 1997]	1980-1984	55%
[Pressman, 1993]	Años 80	60%
[Rock-Evans y Hales, 1990]	1987	67%
[Schach, 1990]	1987	67%
[Pigoski, 1997]	1985-1989	75%
[Frazer, 1992]	1990	80%
[Pressman, 1993]	Años 90 (prev.)	90%

Costes del MS (ii)



Costes del MS (iii)

- Existen empresas que se acercan a porcentajes del 95% de los recursos dedicados al mantenimiento, con lo cual se hace imposible el desarrollo de nuevos productos software. Esta situación se conoce como *Barrera de Mantenimiento*.
- En general, el porcentaje de recursos necesarios para mantenimiento se incrementa a medida que se produce más software (ver figura).
- Los estudios sobre la situación del mercado comercial del MS son relativamente escasos:
 - En España, el estudio del Ministerio de Industria y Energía sobre las Tecnologías de la Información calcula que en 1996 el mantenimiento del software supuso 31.184 millones de pesetas, a los que habría que añadir una parte importante de los 88.140 millones reseñados en el epígrafe de externalización (outsourcing).

El MS supone, cada vez más, un mercado importantísimo.

El efecto *Iceberg*: costes intangibles (i)

- Cuando se planifican los costes de mantenimiento, los analistas-programadores experimentados tienen la impresión de que el MS es algo descontrolado y que nunca se sabe qué va a pasar (es algo así como predecir el futuro). Parece como si fuese un *iceberg* del cual sólo se percibe una pequeña parte, pero bajo cuya superficie se esconde una gran cantidad de problemas potenciales y de costes encubiertos.
- En la parte sumergida de este iceberg se ocultan otros costes, menos tangibles que los monetarios, pero que pueden ser causa de muchas preocupaciones.
- Un coste intangible del MS se encuentra en las oportunidades de desarrollo que se han de posponer o que se pierden, debido a que los recursos disponibles están dedicados a las tareas de mantenimiento.

El efecto *Iceberg*: costes intangibles (ii)

- Otros costes intangibles son los siguientes:
 - Insatisfacción del cliente cuando no se puede atender en un tiempo aceptable una petición de reparación o modificación que parece razonable.
 - Los errores ocultos introducidos al cambiar el software durante el mantenimiento reducen la calidad global del producto.
 - Perjuicio en otros proyectos de desarrollo cuando la plantilla tiene que dejarlos, total o parcialmente, para atender peticiones de mantenimiento.
- En suma, un coste final del mantenimiento del software es la **reducción que se produce en la productividad de los informáticos** al iniciar el mantenimiento de aplicaciones antiguas.
- Algunos estudios han calculado reducciones de la productividad - medida en LDC por persona y mes - de 40 a 1, es decir, el coste de mantener (modificar) una línea de código puede llegar a ser 40 veces más alto que el de escribirla durante el proceso de desarrollo.

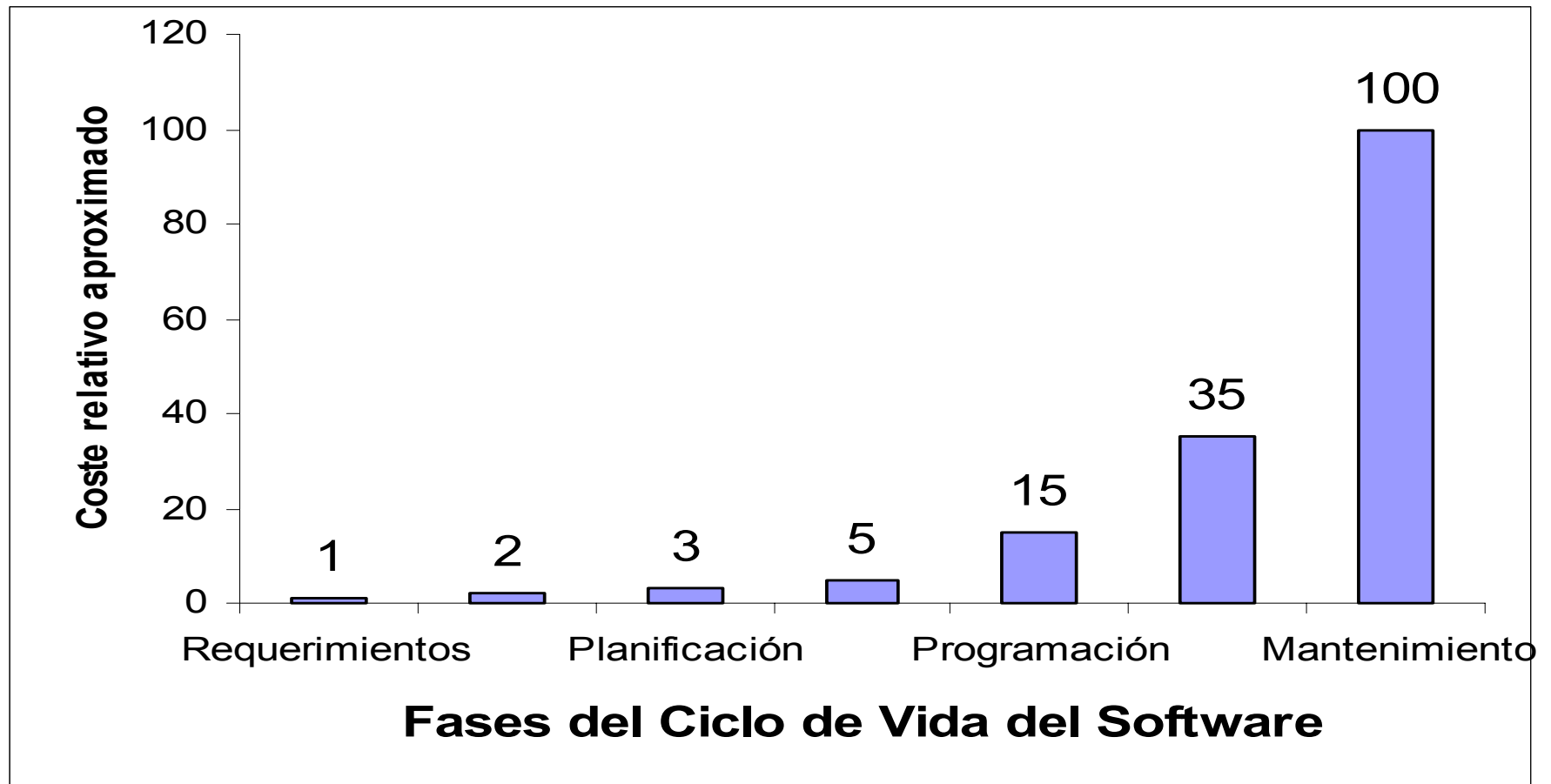
Causas del alto coste del MS (i)

- Son varias las causas de que en la mayoría de las organizaciones actuales se requiera mucho trabajo de mantenimiento:
 - Una gran cantidad del software que existe actualmente ha sido desarrollado hace más de 10 años. Aunque estos programas fuesen creados utilizando las mejores técnicas de diseño y codificación existentes en su momento (la mayoría no lo fueron), se construyeron con restricciones de tamaño y espacio de almacenamiento y se desarrollaron con herramientas tecnológicamente desfasadas.
 - Estos programas han sufrido una o varias migraciones a nuevas plataformas o sistemas operativos.
 - Y han experimentado múltiples modificaciones para mejorarlos y adaptarlos a las nuevas necesidades de los usuarios.
 - Todos estos cambios se realizaron sin tener en cuenta la arquitectura general del sistema (no se aplicaron técnicas de ingeniería inversa o reingeniería).

Causas del alto coste del MS (ii)

- El resultado de todo lo anterior, es la existencia de sistemas software con una baja calidad:
 - diseño pobre de las estructuras de datos,
 - mala codificación,
 - lógica defectuosa, y
 - documentación escasa.
- Pero que tienen que seguir funcionando, y por tanto, tienen que ser mantenidos
 - baja calidad => mayores costes de mantenimiento.
- Otra causa directa de los grandes costes del MS es que el coste relativo de reparar un defecto aumenta considerablemente en las últimas etapas del ciclo de vida del software, de forma que la relación entre el coste de detectar y reparar un defecto en la fase de análisis de requisitos y en la fase de mantenimiento es de 1 a 100 respectivamente (ver figura).

Causas del alto coste del MS (iii)



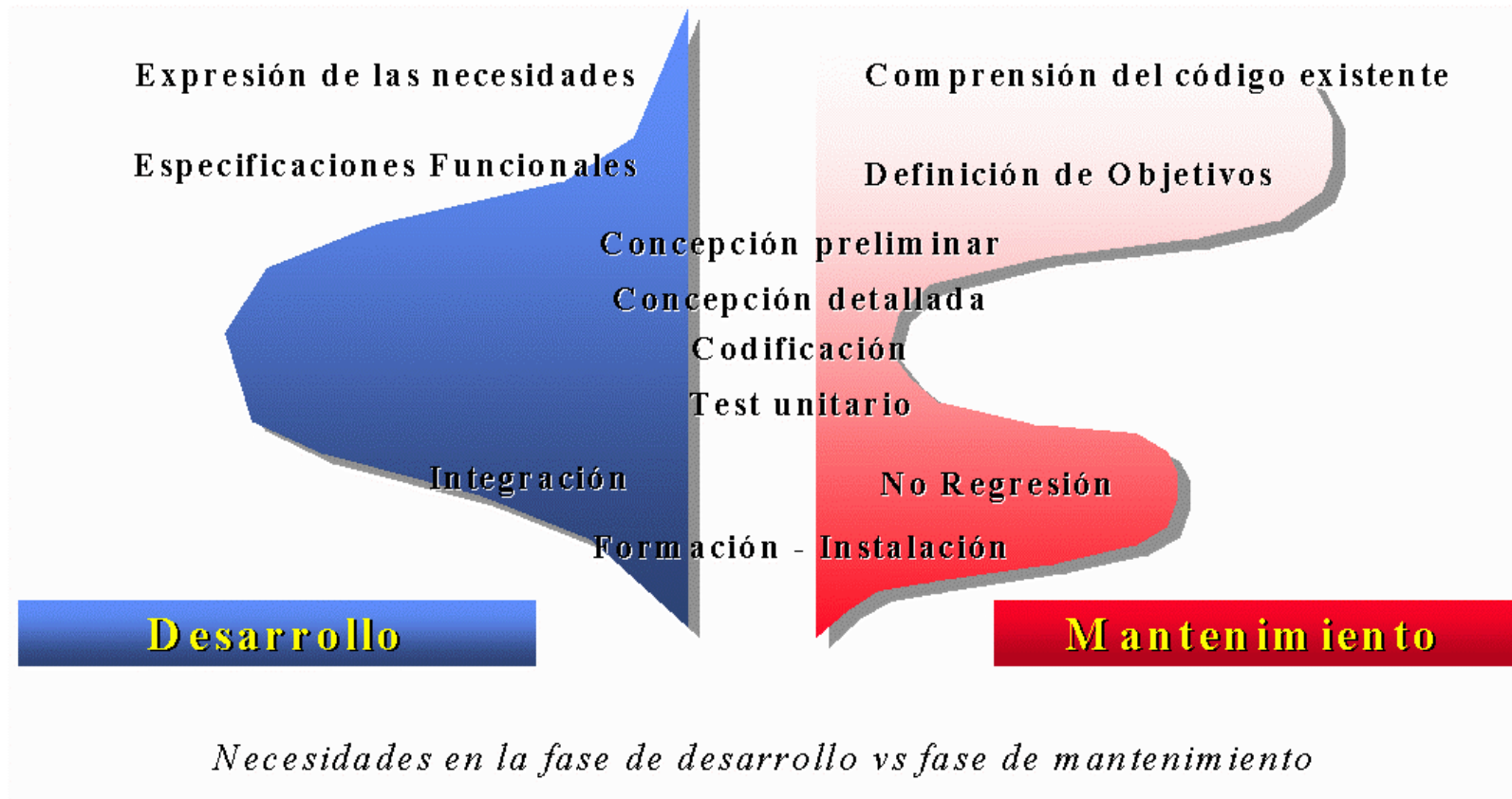
Coste relativo aproximado de detectar y corregir defectos.

Causas del alto coste del MS (iv)

- Algunas de las razones por las que es menos costoso detectar y corregir un error durante las etapas iniciales del ciclo de vida que durante las etapas últimas son:
 - Es más fácil cambiar la documentación (por ejemplo, los documentos de especificación o de diseño) que modificar el código.
 - Un cambio durante una fase tardía puede requerir que sea modificada la documentación de todas las fases anteriores.
 - Es más fácil encontrar un defecto durante la fase en la cual se ha introducido el defecto que tratar de detectar y corregir los efectos provocados por el defecto en una fase posterior.
 - La causa de un defecto puede esconderse en la inexistencia o falta de actualización de los documentos de especificación o diseño.

Causas del alto coste del MS (v)

- Los costes del mantenimiento se incrementan al utilizar técnicas y metodologías poco actas, casi siempre pensadas para las fases previas del ciclo de vida (ver figura).

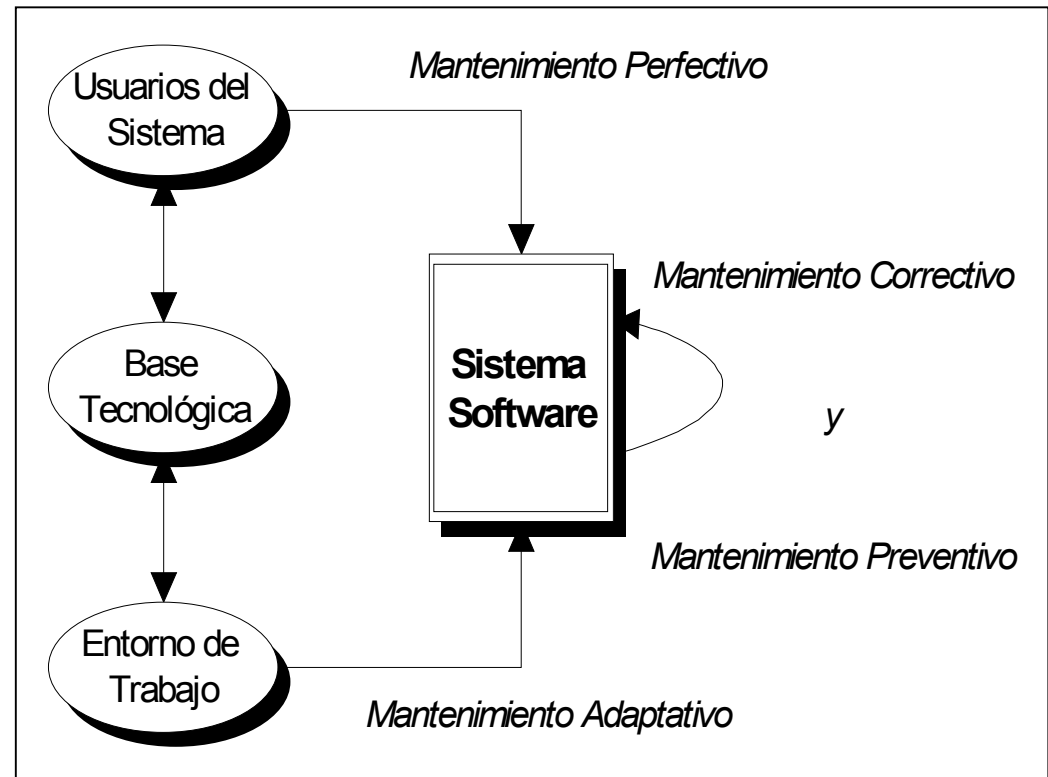


Tipos de Mantenimiento (i)

- En la definición de mantenimiento aparecen indicados, directa o indirectamente, cuatro tipos de mantenimiento:
 - Corregir defectos → *correctivo*
 - Mejorar el rendimiento → *preventivo/perfectivo*
u otras propiedades
 - Adaptar a un cambio de entorno → *adaptativo*
- Las definiciones de ISO e IEEE no coinciden.
- En algunas propuestas más modernas, se proponen algunas subdivisiones (metodología MANTEMA).

Tipos de Mantenimiento (ii)

- Un resumen del papel que representa cada tipo de mantenimiento aparece en la figura:
 - Mientras que el cambio tecnológico afecta indirectamente a los sistemas software, el entorno de trabajo y los usuarios lo hacen directamente, produciendo demandas de mantenimiento adaptativo y perfectivo respectivamente.



Tipos de Mantenimiento (iii)

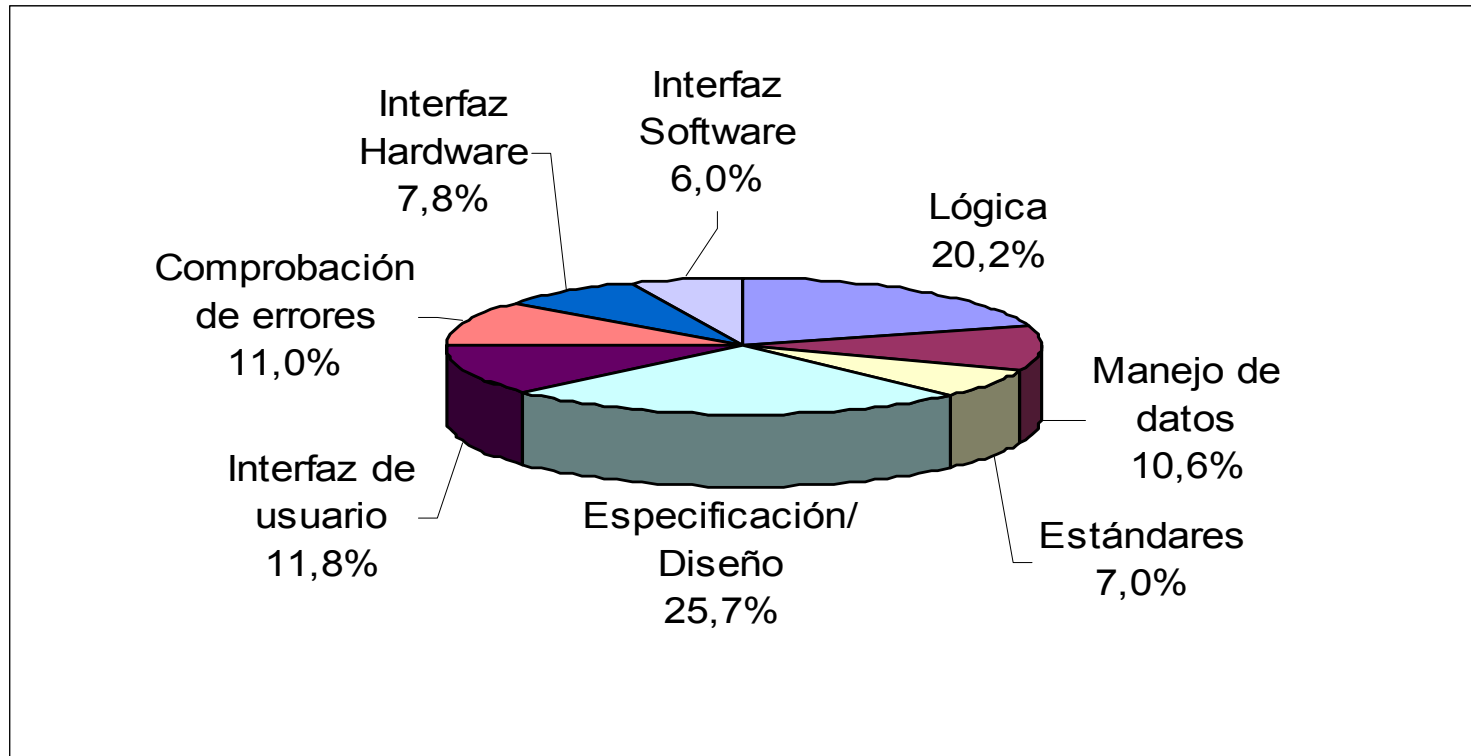
- En **MANTEMA** se trabaja con los siguientes tipos:
 - No Planificable (NP):
 - Correctivo Urgente (UC): localizar y eliminar los posibles defectos que bloquean el programa o los procesos de funcionamiento de la empresa.
 - Planificable (P):
 - Correctivo No Urgente (NUC): localizar y eliminar los posibles defectos de los programas que no son bloqueantes.
 - Perfectivo (PER): añadir al software nuevas funcionalidades solicitadas por los usuarios.
 - Adaptativo (A): modificar el software para adaptarlo a cambios en el entorno de trabajo (hardware o software).
 - Preventivo (PRE): modificar el software para mejorar sus propiedades (calidad, mantenibilidad, etc.).

Mantenimiento Correctivo (i)

- A pesar de las pruebas y verificaciones que aparecen en etapas anteriores del ciclo de vida del software, los programas pueden tener defectos. El mantenimiento correctivo tiene por objetivo *localizar y eliminar los posibles defectos* de los programas.
- Un defecto en un sistema es una característica del sistema con el potencial de causar un fallo.
- Un *fallo* ocurre cuando el comportamiento de un sistema es diferente del establecido en la especificación. Entre otros, los fallos en el software pueden ser de:
 - Procesamiento, por ejemplo, salidas incorrectas de un programa.
 - Rendimiento, por ejemplo, tiempo de respuesta demasiado alto en una búsqueda de información.
 - Programación, por ejemplo, inconsistencias en el diseño de un programa.
 - Documentación, por ejemplo, inconsistencias entre la funcionalidad de un programa y el manual de usuario.

Mantenimiento Correctivo (ii)

- En la figura se muestra una distribución de las causas de los defectos según un estudio estadístico realizado en 1994.



Origen de los defectos del software

Mantenimiento Adaptativo (i)

- Este tipo de mantenimiento consiste en la modificación de un programa debido a *cambios en el entorno* (hardware o software) en el cual se ejecuta.
- Los cambios pueden afectar a:
 - el sistema operativo (cambio a uno más moderno),
 - la arquitectura física del sistema informático (paso de una arquitectura de red de área local a Internet/Intranet),
 - o al entorno de desarrollo del software (incorporación de nuevos elementos o herramientas como ODBC).
- La envergadura del cambio necesario puede ser muy diferente: desde un pequeño retoque en la estructura de un módulo hasta tener que reescribir prácticamente todo el programa para su ejecución en un ambiente distribuido en una red.

Mantenimiento Adaptativo (ii)

- Los cambios en el entorno software pueden ser de dos clases:
 - En el entorno de los *datos*, por ejemplo, al dejar de trabajar con un sistema de ficheros clásico y sustituirlo por un sistema de gestión de bases de datos relacionales.
 - En el entorno de los *procesos*, por ejemplo, migrando a una nueva plataforma de desarrollo con componentes distribuidos, Java, ActiveX, etc.
- Este tipo de mantenimiento es cada vez más frecuente debido principalmente al cambio, cada vez más rápido, en los diversos aspectos de la informática: nuevas generaciones de hardware, nuevos sistemas operativos -o versiones de los antiguos-, y mejoras en los periféricos o en otros elementos del sistema (frente a esto, la vida útil de un sistema software puede superar fácilmente los diez años).

Mantenimiento Perfectivo

- Cambios en la especificación, normalmente debidos a cambios en los requerimientos de un producto software, implican un nuevo tipo de mantenimiento llamado perfectivo. La casuística es muy variada. Desde algo tan simple como cambiar el formato de impresión de un informe, hasta la incorporación de un nuevo módulo funcional. Podemos definir el mantenimiento perfectivo como el conjunto de actividades para *mejorar o añadir nuevas funcionalidades* requeridas por el usuario.
- Algunos autores dividen este tipo de mantenimiento en dos:
 - Mantenimiento de Ampliación: orientado a la incorporación de nuevas funcionalidades.
 - Mantenimiento de Eficiencia: que busca la mejora de la eficiencia de ejecución.

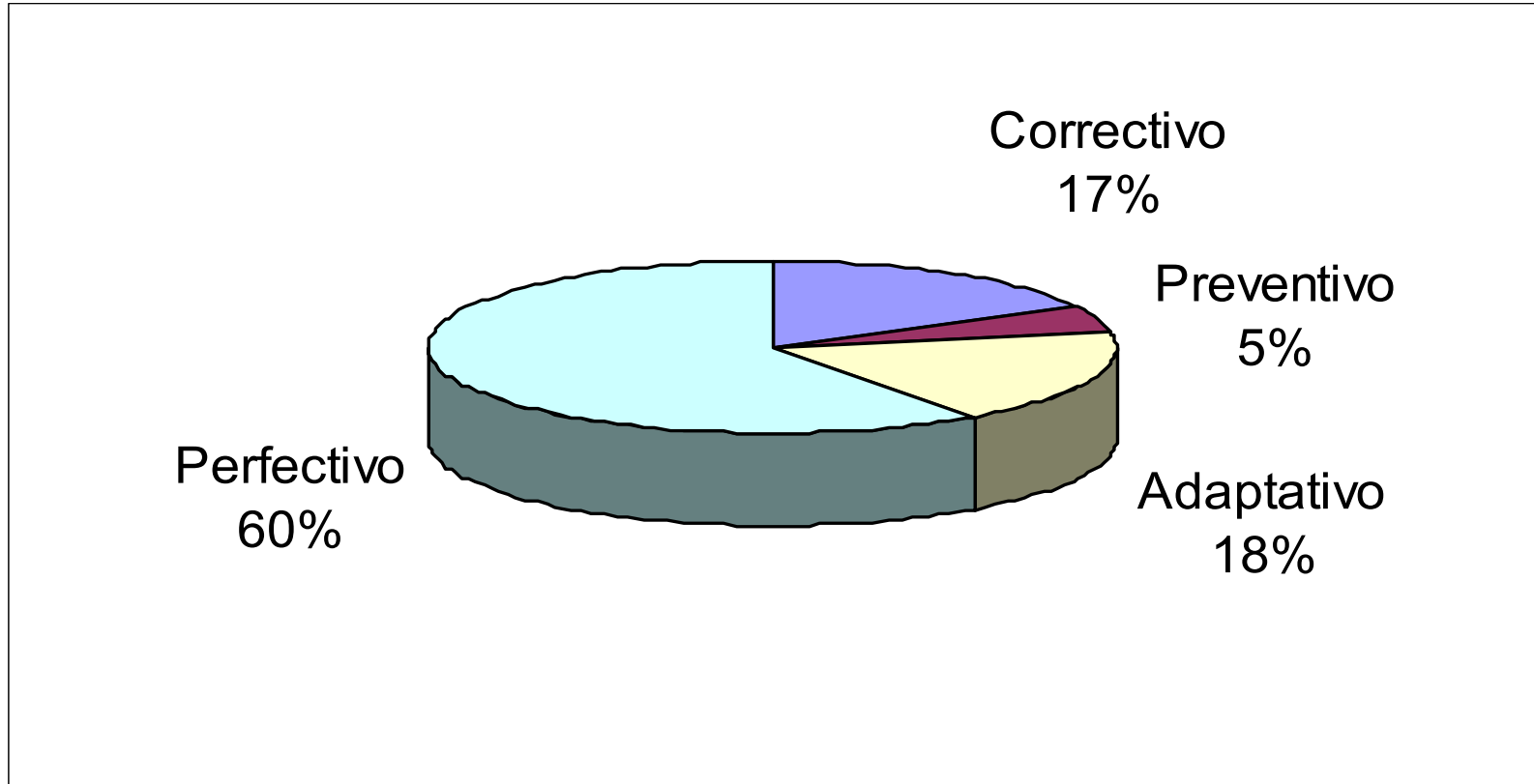
Mantenimiento Preventivo

- Este último tipo de mantenimiento consiste en la modificación del software para **mejorar las propiedades** de dicho software (por ejemplo, aumentando su calidad y/o su mantenibilidad) sin alterar sus especificaciones funcionales. Algunas maneras de hacerlo son:
 - incluir sentencias que comprueben la validez de los datos de entrada,
 - reestructurar los programas para mejorar su legibilidad, o
 - incluir nuevos comentarios que faciliten la posterior comprensión del programa.
- En algunos casos se ha planteado el Mantenimiento para la Reutilización, consistente en modificar el software (buscando y modificando componentes para incluirlos en bibliotecas) para que sea mas fácilmente reutilizable. En realidad este tipo de mantenimiento es preventivo, especializado en mejorar la propiedad de reusabilidad del software.

Actividades del MS (i)

- El desconocimiento de las actividades que implica el MS puede inducir a minusvalorar su importancia.
- Lo primero que se suele asociar con el MS es la corrección de errores de los programas. Por esta causa, la impresión mas generalizada entre los gestores, usuarios, e incluso entre los propios informáticos, es que la mayor parte del mantenimiento que se realiza en el mundo es de tipo correctivo.
- Sin embargo, los principales estudios realizados sobre el tema indican que esta impresión es equivocada, y establecen que *el mantenimiento perfectivo es el tipo más habitual* (ver figura).
- El establecimiento de analogías entre el MS y el mantenimiento del hardware puede conducir a confusión, ya que el software, a diferencia del hardware, no se desgasta y, por tanto, la principal actividad asociada con el mantenimiento del hardware -reemplazar o reparar las piezas estropeadas o defectuosas- no es aplicable al software.

Actividades del MS (ii)



Esfuerzo dedicado a cada tipo de mantenimiento

Actividades del MS (iii)

- Las actividades de MS se pueden agrupar en *tres categorías* funcionales:
 - **Comprensión del software y de los cambios a realizar:** para poder modificar un programa, los programadores necesitan conocer su funcionalidad y objetivos, su estructura interna y los requisitos de operación. De no ser así, se corre un gran riesgo de introducir nuevos defectos que en el futuro supondrán un coste de mantenimiento adicional.
 - **Modificación del software:** para incorporar los cambios necesarios se deben crear y modificar las estructuras de datos, la lógica de los procesos, las interfaces y la documentación. Los programadores deben conocer lo mejor posible las repercusiones que tienen en el sistema los cambios que están realizando, con el fin de evitar al máximo posible los efectos secundarios.
 - **Realización de pruebas:** para validar los cambios se deben realizar pruebas selectivas que nos permitan comprobar la corrección del software. Esta actividad es necesaria siempre, ya que incluso un cambio muy pequeño no verificado puede producir defectos en el software que reduzcan su calidad y fiabilidad.

Actividades del MS (iv)

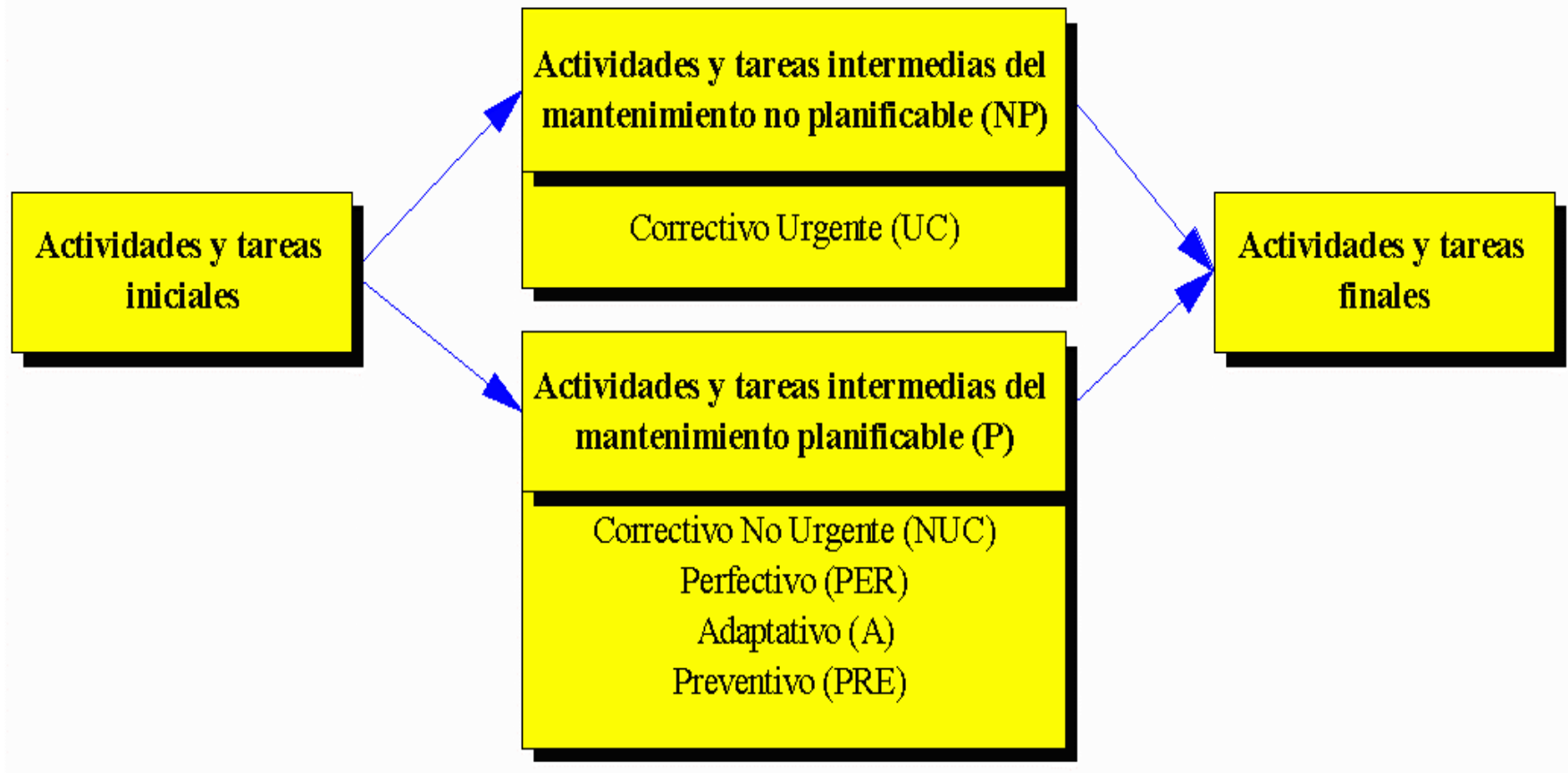
- En la tabla se indican las proporciones del tiempo de mantenimiento que supone cada actividad específica según McClure [1992]:

Categoría	Actividad	% Tiempo
a) Comprensión del software y de los cambios a realizar	Estudiar las peticiones	18%
	Estudiar la documentación	6%
	Estudiar el código	23%
b) Modificación del software	Modificar el código	19%
	Actualizar la documentación	6%
c) Realización de pruebas	Diseñar y realizar pruebas	28%

Actividades según el Tipo de Mantenimiento (i)

- Según el tipo de mantenimiento, se deberán realizar un conjunto de actividades y tareas diferentes.
- El mantenimiento UC (MANTEMA) tiene diferencias claras con los demás tipos debido a que la urgencia necesaria para reparar los errores no permite realizar ninguna actividad de planificación.
- Se pueden definir subconjuntos de actividades y tareas comunes a varios tipos de mantenimiento, cumpliéndose que:
 - Existe un conjunto de actividades y tareas "*iniciales*" comunes a todos los tipos de mantenimiento.
 - Existe un conjunto de actividades y tareas "*finales*" comunes a todos los tipos de mantenimiento.
 - Las actividades "*intermedias*" de los tipos de mantenimiento P son bastante coincidentes.

Actividades según el Tipo de Mantenimiento (ii)



Tipos de mantenimiento y actividades en MANTEMA