

Arquitectura de Aplicaciones

Contenidos del capítulo

- 13.1 Sistemas de procesamiento de datos
- 13.2 Sistemas de procesamiento de transacciones
- 13.3 Sistemas de procesamiento de eventos
- 13.4 Sistemas de procesamiento de lenguajes

Normalmente, los sistemas del mismo tipo tienen arquitecturas similares, y la diferencia entre estos sistemas está en la funcionalidad detallada que proporcionan. Un sistema genérico se configura y adapta para crear una aplicación específica de negocio.

Como diseñador de software, usted puede usar arquitecturas de aplicaciones genéricas de varias formas:

1. Como un punto de partida para el proceso de diseño arquitectónico: puede basar sus diseños iniciales sobre arquitecturas genéricas.
2. Como una lista de comprobación de un diseño: puede utilizarse la arquitectura de otro proyecto para compararlos.
3. Como una forma de organizar el trabajo del grupo de desarrollo. La arquitectura de la aplicación identifica características estructurales estables de las arquitecturas de los sistemas y, en muchos casos, es posible desarrollarlas en paralelo. Usted puede asignar trabajo a miembros del grupo para implementar diferentes subsistemas dentro de la arquitectura.
4. Como una forma de evaluar los componentes para su reutilización: reutilizando componentes existentes.
5. Como un vocabulario para hablar sobre tipos de aplicaciones: usar los conceptos identificados en una arquitectura genérica.

Hay muchos tipos de sistemas de aplicaciones, que tienen muchas cosas en común, a continuación se describen las arquitecturas de 4 grandes tipos de aplicaciones:

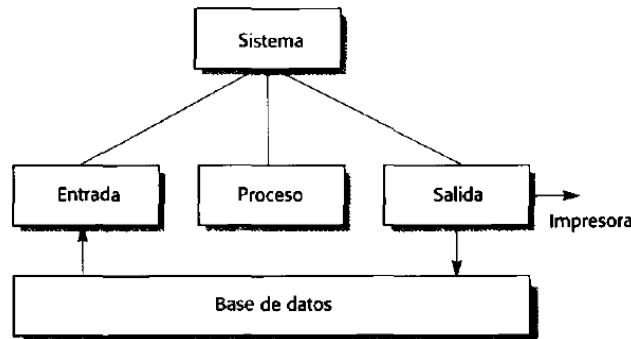
1. **Aplicaciones de procesamiento de datos**. Las aplicaciones de procesamiento de datos son aplicaciones conducidas por los datos. Procesan datos por lotes sin intervenciones explícitas del usuario durante el procesamiento. Las acciones específicas tomadas por la aplicación dependen de los datos que se están procesando. Los sistemas de procesamiento por lotes se usan normalmente en aplicaciones de negocio en donde se realizan operaciones similares sobre grandes cantidades de datos. Dichas aplicaciones manejan un amplio rango de funciones administrativas tales como nóminas, facturación, contabilidad y publicidad.
2. **Aplicaciones de procesamiento de transacciones**. Las aplicaciones de procesamiento de transacciones son aplicaciones centradas en bases de datos que procesan peticiones del usuario para obtener información y para actualizar la información en una base de datos. Éstos son los tipos más comunes de sistemas de negocio interactivos. Se organizan de forma que las acciones del usuario no pueden interferir entre sí y se mantenga la integridad de la base de datos.
3. **Sistemas de procesamiento de eventos**. Las que las acciones del sistema dependen de la interpretación de eventos en el entorno del sistema. Estos eventos podrían ser la entrada de una orden por un usuario del sistema o un cambio en las variables que son monitorizadas por el sistema.
4. **Sistemas de procesamiento de lenguajes**. Las intenciones del usuario se expresan en un lenguaje formal (como, por ejemplo, Java), procesan este lenguaje en algún formato interno y entonces interpretan su representación interna. Los más conocidos son los compiladores, lenguajes se utilizan para interpretar lenguajes de comandos para bases de datos, etc.

Los sistemas de negocio son generalmente sistemas de procesamiento de transacciones o de datos, y la mayoría del software de computadoras personales se construye sobre una arquitectura de procesamiento de eventos. Todo el desarrollo del software se centra en los sistemas de procesamiento de lenguajes tales como compiladores. Los sistemas de procesamiento por lotes y procesamiento de transacciones se centran en bases de datos. En las aplicaciones complejas la arquitectura es la mayoría de las veces un híbrido.

13.1 Sistemas de procesamiento de datos

Los sistemas de procesamiento de datos son sistemas de procesamiento por lotes en los que los datos son introducidos y extraídos por lotes a partir de un fichero o base de datos en lugar de ser introducidos y extraídos por un terminal de usuario.

La arquitectura de los sistemas de procesamiento por lotes tiene tres componentes principales, como se observa en la siguiente figura:



Un modelo de entrada-proceso-salida de un sistema de procesamiento de datos.

Un componente de entrada reúne entradas desde una o más fuentes. Un componente de procesamiento realiza cálculos utilizando estas entradas; y un componente de salida genera salidas para ser escritas en la base de datos e impresoras.

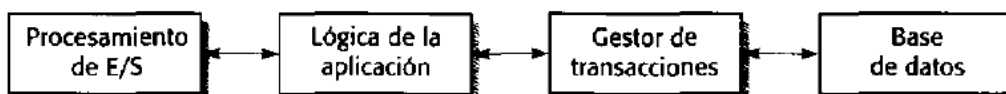
La naturaleza de los sistemas de procesamiento de datos, en donde los registros o transacciones se procesan en serie sin necesidad de mantener el estado entre las transacciones, implica que estos sistemas sean naturalmente orientados a funciones en vez de orientados a objetos. Las funciones son componentes que no mantienen información del estado interno de una invocación a otra.

Los diagramas de flujos de datos constituyen una forma de representar sistemas orientados a funciones en donde cada rectángulo con bordes redondeados en el flujo de datos representa una función que implementa algunas transformaciones de datos, y cada flecha representa un elemento de datos procesado por la función. Los ficheros o almacenes de datos se representan como rectángulos. La ventaja de los diagramas de flujo de datos es que muestran el procesamiento de los datos desde su entrada hasta su salida.

13.2 Sistemas de procesamiento de transacciones

Los sistemas de procesamiento de transacciones se diseñan para procesar peticiones de usuario a fin de obtener información de una base de datos o peticiones para actualizar la base de datos. Técnicamente, una transacción de una base de datos es una secuencia de operaciones tratada como una única unidad (una unidad atómica). Todas las operaciones de una transacción tienen que ser completadas antes de que los cambios en la base de datos sean permanentes. Desde la perspectiva de un usuario, una transacción es cualquier secuencia coherente de operaciones que satisface un objetivo.

En la siguiente figura observamos la estructura de las aplicaciones de procesamiento de transacciones.

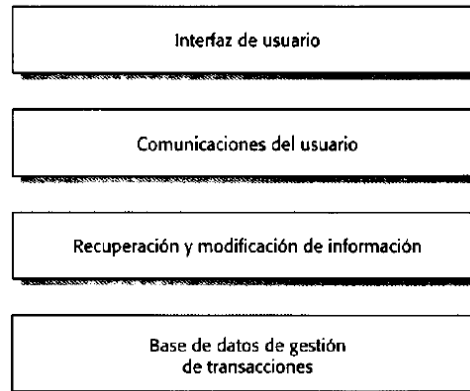


Los sistemas de procesamiento de transacciones son normalmente sistemas interactivos en donde los usuarios realizan peticiones de servicios de forma asíncrona. Primero un usuario realiza una petición al sistema a través de un componente de procesamiento de E/S. La petición se procesa por alguna lógica específica de la aplicación. Una transacción se crea y se envía a un gestor de transacciones, el cual normalmente está embebido en el sistema de gestión de base de datos. Después de que el gestor de transacciones haya asegurado que la transacción se ha realizado correctamente, envía una señal a la aplicación indicando que el procesamiento ha finalizado.

La estructura entrada-procesa-salida que podemos observar en las aplicaciones de procesamiento de datos también se aplican a muchos sistemas de procesamiento de transacciones.

13.2.1 Sistemas de información y de gestión de recursos

Todos los sistemas que implican una interacción con una base de datos compartida pueden considerarse como sistemas de información basados en transacciones. El desarrollo de la WWW significa que un enorme número de sistemas de información pasaron de ser sistemas organizacionales especializados a ser sistemas de propósito general universalmente accesibles.



Modelo por capas de un sistema de información

La Figura anterior es un modelo muy general de un sistema de información. El sistema se modela utilizando una aproximación de máquina abstracta o por capas, en donde la capa superior soporta la interfaz de usuario y la capa inferior la base de datos del sistema. La capa de comunicaciones con el usuario maneja todas las entradas y salidas de la interfaz de usuario, y la capa de recuperación de información incluye lógica específica de la aplicación para acceder y actualizar la base de datos.

Podemos observar la misma estructura genérica de cuatro capas en otros tipos de sistemas de información, como por ejemplo en los sistemas de asignación de recursos, que gestionan una cantidad fija de algún recurso determinado. Los componentes de un sistema de asignación de recursos comprenden:

1. Una base de datos de recurso.
2. Un conjunto de reglas que describe las reglas para la asignación de recursos.
3. Un componente de gestión de recursos que permite al suministrador de los recursos añadir, editar o borrar recursos del sistema.
4. Un componente de asignación de recursos.
5. Un módulo de autenticación de usuarios que permite al sistema comprobar qué recursos están siendo asignados a un usuario acreditado.
6. Un módulo de gestión de consultas que permite a los usuarios consultar qué recursos están disponibles.
7. Un componente de entrega de recursos que prepara los recursos para su entrega al solicitante.
8. Un componente de interfaz de usuario (a menudo un navegador web) que está fuera del sistema y permite al solicitante del recurso realizar consultas y peticiones para el recurso que se va a asignar.

13.3 Sistemas de procesamiento de eventos

Los sistemas de procesamiento de eventos responden a eventos en el entorno del sistema o interfaz del usuario. La principal característica de los sistemas de procesamiento de eventos es que la llegada de los eventos es impredecible y el sistema debe ser capaz de tratar estos eventos cuando ocurren.

Todos nosotros usamos sistemas basados en eventos como éstos en nuestras propias computadoras - procesadores de texto, sistemas de presentaciones y juegos están todos conducidos por eventos desde la interfaz de usuario-.

Los sistemas de tiempo real, los cuales realizan acciones en «tiempo real» como respuesta a algunos estímulos externos, son también sistemas de procesamiento basados en eventos.

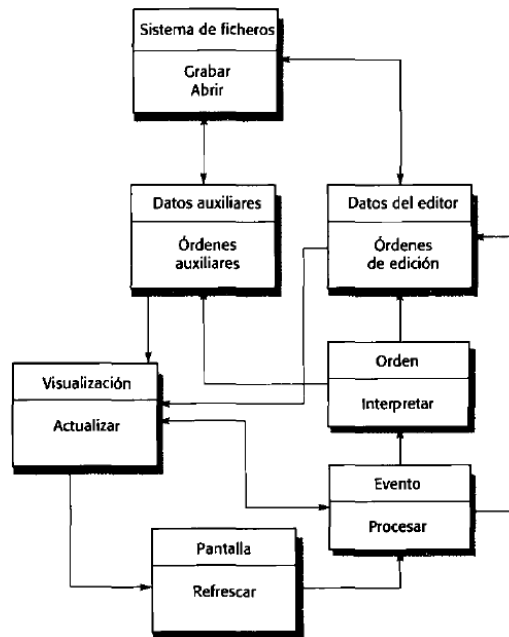
Sin embargo, para sistemas de tiempo real, los eventos no son normalmente eventos de interfaz de usuario, sino eventos asociados con servidores o actuadores en el sistema. Debido a la necesidad de respuesta en tiempo real a eventos no predecibles, estos sistemas de tiempo real se organizan normalmente como un conjunto de procesos cooperativos.

Nos centraremos en la descripción de la arquitectura genérica de los sistemas de edición.

Los sistemas de edición tienen varias características que los distinguen de otros tipos de sistemas y que influyen en su diseño arquitectónico:

1. Los sistemas de edición son principalmente sistemas para un único usuario. Por lo tanto, no tienen que tratar los problemas de múltiples accesos concurrentes a los datos y tienen una gestión de los datos más sencilla que los sistemas basados en transacciones.
2. Tienen que proporcionar una rápida realimentación de las acciones del usuario tales como «seleccionar» y «borrar». Esto significa que tienen que funcionar con representaciones de los datos que se almacenan en la memoria de la computadora en vez de en el disco. Debido a que los datos están en la memoria volátil, pueden perderse si hay un fallo en el sistema; por lo tanto, los sistemas de edición deberían realizar alguna previsión para la recuperación de errores.
3. Muchos sistemas de edición incluyen facilidades de recuperación que automáticamente guardan el trabajo en curso y recuperan el trabajo para el usuario en el caso de que se produzca un fallo en el sistema.

Las responsabilidades de los componentes arquitectónicos son:



Un modelo arquitectónico de un sistema de edición. (Conjunto de objetos que interactúan)

1. Pantalla. Este objeto monitoriza el segmento de memoria de la pantalla y detecta la ocurrencia de eventos. A continuación, estos eventos se envían al objeto de procesamiento de eventos junto con sus coordenadas de pantalla.

2. Evento. Este objeto es disparado por un evento originado desde la Pantalla. Utiliza el conocimiento sobre lo que se está visualizando para interpretar este evento y traducirlo al comando de edición adecuado. A continuación, este comando se envía al objeto responsable de la interpretación de comandos. Para los eventos más frecuentes, tales como pulsaciones de ratón o pulsaciones de teclas, el objeto evento puede comunicarse directamente con la estructura de datos. Esto permite actualizaciones más rápidas de dicha estructura.

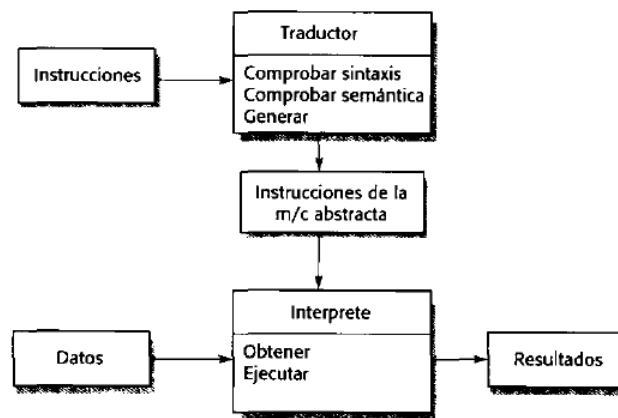
3. Orden. Este objeto procesa un comando que proviene del objeto evento y realiza una llamada al método adecuado en el objeto Editor de datos para ejecutar el comando.

4. Editor de datos. Cuando se llama al comando adecuado en el objeto Editor de datos, se actualiza la estructura de datos y se llama al método Actualizar en Visualización para visualizar los datos modificados.
5. Datos auxiliares. Además de la propia estructura de datos, los editores gestionan otros datos tales como estilos y preferencias.
6. Sistema de ficheros. Este objeto maneja todas las operaciones para abrir y guardar ficheros. Éstos pueden ser ficheros de datos auxiliares o ficheros del editor de datos.
7. Visualización. Este objeto mantiene un seguimiento de la organización de lo que se visualiza en la pantalla. Realiza una llamada al método Refrescar del objeto Pantalla cuando el contenido de la pantalla ha sido modificado.

13.4 Sistemas de procesamiento de lenguajes

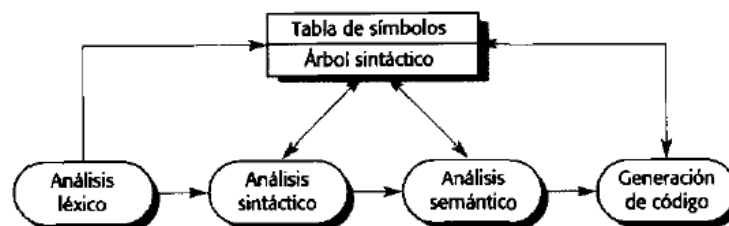
Los sistemas de procesamiento de lenguajes aceptan lenguaje natural o artificial como entrada y generan alguna otra representación de ese lenguaje como salida. En ingeniería del software, los sistemas de procesamiento de lenguajes más extensamente usados son los compiladores, que traducen un lenguaje de programación artificial de alto nivel a código máquina, pero otros sistemas de procesamiento de lenguajes traducen una descripción de datos en XML a comandos de consulta de una base de datos y sistemas de procesamiento de lenguaje natural que intentan traducir un lenguaje natural a otro.

En la siguiente figura se ilustra la arquitectura de un sistema de procesamiento de lenguajes en su nivel más abstracto.



Las instrucciones describen lo que tiene que realizarse y tienen que traducirse por un traductor a algún formato interno. Las instrucciones se corresponden con las instrucciones máquina para una máquina abstracta. Estas instrucciones son interpretadas por otro componente que obtiene las instrucciones para su ejecución y las ejecuta usando, si es necesario, datos del entorno. La salida del proceso es el resultado de interpretar las instrucciones sobre los datos de entrada.

Los sistemas de procesamiento de lenguajes se usan en situaciones en las que la forma más fácil de resolver un problema es especificar esa solución como un algoritmo o como una descripción de los datos del sistema. Los traductores en un sistema de procesamiento de lenguajes tienen una arquitectura genérica (Ver Figura).



Incluye los siguientes componentes:

1. Un analizador léxico, que toma como entrada los símbolos del lenguaje y los convierte a un formato interno.

2. Una tabla de símbolos, que almacena información sobre los nombres de las entidades usadas en el texto que se está traduciendo.
3. Un analizador sintáctico, que comprueba la sintaxis del lenguaje que se está traduciendo. Utiliza una gramática definida del lenguaje y construye un árbol sintáctico.
4. Un árbol sintáctico, que es una estructura interna que representa al programa que se está compilando.
5. Un analizador semántico, que utiliza información del árbol sintáctico y de la tabla de símbolos para comprobar la corrección semántica del texto en el lenguaje de entrada.
6. Un generador de código, que «recorre» el árbol sintáctico y genera código de máquina abstracta.

Los componentes que forman un sistema de procesamiento de lenguajes pueden organizarse de acuerdo con diferentes modelos arquitectónicos. Los compiladores pueden implementarse utilizando un modelo compuesto. Puede utilizarse una arquitectura de flujo de datos con la tabla de símbolos actuando como un repositorio para datos compartidos.

Este modelo de compilación de flujo de datos todavía se usa en general. Es efectivo en entornos de procesamiento por lotes en donde los programas son compilados y ejecutados sin la interacción del usuario.

PUNTOS CLAVE

- Los modelos genéricos de las arquitecturas de sistemas de aplicaciones nos ayudan a entender el funcionamiento de las aplicaciones, comparar aplicaciones del mismo tipo, validar los diseños de los sistemas de aplicaciones y evaluar componentes a gran escala para su reutilización.
- Muchas aplicaciones pertenecen a una de las cuatro clases de aplicaciones genéricas o son combinaciones de estas aplicaciones genéricas. Los cuatro tipos de aplicaciones genéricas explicadas aquí son sistemas de procesamiento de datos, sistemas de procesamiento de transacciones, sistemas de procesamiento de eventos y sistemas de procesamiento de lenguajes.
- Los sistemas de procesamiento de datos operan en modo diferido y generalmente tienen una estructura entrada-proceso-salida. Los registros se introducen en el sistema, la información se procesa y se generan las salidas.
- Los sistemas de procesamiento de transacciones son sistemas interactivos que permiten acceder y modificar la información de una base de datos de forma remota por varios usuarios. Los sistemas de información y los sistemas de gestión de recursos son ejemplos de sistemas de procesamiento de transacciones.
- Los sistemas de procesamiento de eventos incluyen a los sistemas de edición y a los sistemas de tiempo real. En un sistema de edición, se interpretan los eventos de la interfaz de usuario y se modifica una estructura de datos de entrada. Los procesadores de texto y los sistemas de presentaciones son ejemplos de sistemas de edición.
- Los sistemas de procesamiento de lenguajes se utilizan para traducir textos de un lenguaje a otro y para llevar a cabo las instrucciones especificadas en los lenguajes de entrada. Estos incluyen un traductor y una máquina abstracta que ejecuta el lenguaje generado.