**Servidores de Aplicaciones**

Juan Daniel Forner, Adrián Marqués, Miguel Villaescusa y Darío Poves

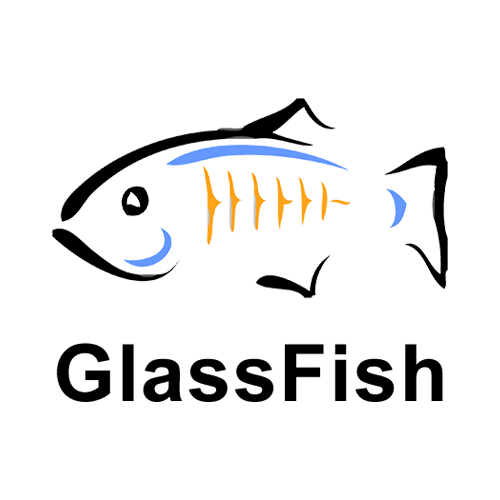
* **Que es un servidor de aplicaciones**

Un servidor de aplicaciones, en el ámbito de la informática, es un componente de software ubicado en una red de ordenadores que ejecuta y gestiona diversas aplicaciones. Este dispositivo software se encarga de proporcionar servicios de aplicación a las computadoras cliente, centralizando y simplificando la lógica de negociación y el acceso a datos en el desarrollo de aplicaciones. Su función principal es crear un entorno para la ejecución de aplicaciones web, abarcando desde sistemas de gestión de contenido hasta aplicaciones empresariales y de comercio electrónico.

Este servidor de aplicaciones administra la ejecución de programas, procesos y servicios necesarios para el funcionamiento de aplicaciones web complejas. Además de ofrecer contenido web, su tarea principal consiste en facilitar la interacción entre los clientes de usuario final y el código de aplicación del lado del servidor, conocido como lógica empresarial. Esta interacción puede llevarse a cabo a través de diversos protocolos de comunicación y puede tener como clientes la interfaz de usuario final de la aplicación, un navegador web o una aplicación móvil. Los beneficios clave de la tecnología de servidores de aplicaciones incluyen la centralización y la simplificación, lo que contribuye a la eficiencia en el desarrollo de aplicaciones. Algunos ejemplos destacados de servidores de aplicaciones son:

WebSphere Application Server es la implementación por parte de IBM de la plataforma Java™ Platform, Enterprise Edition (Java EE). WebSphere Application Server está disponible en paquetes exclusivos que están diseñados para cumplir una amplia gama de requisitos de cliente. En el centro de cada paquete hay un WebSphere Application Server que proporciona el entorno de ejecución para aplicaciones empresariales.



Apache Tomcat es un servidor de aplicaciones de código abierto que ejecuta servlets Java, representa y entrega páginas web que incluyen código JavaServer Page y actúa como servidor para aplicaciones Java Enterprise Edition (Java EE). Tomcat se publicó en 1998 y es el servidor de aplicaciones Java de código abierto más utilizado.

### 

Glassfish es un servidor de aplicaciones Java EE de código abierto lanzado por Sun Microsystems en 2006 y alojado en la actualidad por Eclipse Foundation (enlace externo a ibm.com). Al igual que la mayoría de los servidores de aplicaciones Java, Glassfish admite servlets Java, Enterprise JavaBeans (EJB) y más, pero también puede funcionar como servidor web y dar servicio a contenido web en respuesta a solicitudes HTTP. 

Para el desarrollo de una aplicación empresarial compleja que requiere de gestión de transacciones y alta disponibilidad, una buena elección sería Oracle WebLogic como servidor de aplicaciones. Esta plataforma proporciona servicios avanzados para la gestión de transacciones, la seguridad robusta y la capacidad para ejecutar aplicaciones Java EE a gran escala.

* **Diferencias con un servidor web**

Un servidor web es un tipo de servidor que ofrece contenido estático, como páginas HTML, archivos, imágenes y videos, en respuesta a las solicitudes de los navegadores web. También se conoce como servidor HTTP y se utiliza para distribuir y entregar contenido en redes internas o en Internet. Su tarea principal es despachar los archivos de una página o aplicación web, como textos e imágenes, utilizando los protocolos HTTP o HTTPS para enviar la información solicitada a los navegadores de los usuarios.

Los servidores de aplicaciones normalmente también pueden ofrecer contenido web, pero su tarea principal es habilitar la interacción entre los clientes de usuario final y el código de aplicación del lado del servidor (el código que representa lo que a menudo se denomina lógica empresarial) para generar y entregar contenido dinámico, como resultados de transacciones, soporte de decisiones o análisis en tiempo real. El cliente de un servidor de aplicaciones puede ser la propia interfaz de usuario final de la aplicación, un navegador web o una aplicación móvil, y la interacción cliente-servidor puede producirse a través de diversos protocolos de comunicación.

De esta forma se sitúa entre los servicios de backend como las bases de datos y el servidor web que se encarga de comunicar con los clientes. En la práctica, sin embargo, se ha desdibujado la línea entre los servidores web y los servidores de aplicaciones, sobre todo porque el navegador web ha surgido como el cliente de aplicaciones preferido, y porqué han aumentado las expectativas de los usuarios y el rendimiento de las aplicaciones web.

* **Características de los servidores de aplicaciones**

Un servidor de aplicaciones brinda diversas ventajas, entre las cuales se incluyen:

* **Compatibilidad:**

Al implementar un servidor de aplicaciones en tu red, garantizas que todos los clientes trabajen con la misma versión de software, asegurando así la compatibilidad de los datos intercambiados y almacenados. La ausencia de un servidor de aplicaciones puede derivar en problemas debido a la falta de actualizaciones o al uso de software obsoleto.

* **Seguridad:**

La introducción de un servidor de aplicaciones puede resultar beneficioso desde el punto de vista de la seguridad. Por ejemplo, evita la conexión directa entre una página web y su base de datos, proporcionando una capa adicional de protección contra ataques de malware como los programas de inyección SQL.

* **Rendimiento:**

Los servidores de aplicaciones pueden mejorar el rendimiento general, especialmente en programas extensos o de uso frecuente. Las opciones de escalado contribuyen a optimizar el rendimiento, al igual que la capacidad de agrupar y controlar el acceso a la base de datos.

* **Configuración:**

La centralización a través de un servidor de aplicaciones implica que la configuración no necesita llevarse a cabo en cada dispositivo individual, simplificando así el proceso y ahorrando tiempo y recursos.

* **Escalabilidad:**

Gracias a las diversas opciones de conexión con la base de datos, es posible lograr una mayor escalabilidad, ofreciendo una mayor flexibilidad para los usuarios. No obstante, es importante mencionar algunos inconvenientes:

* **Costes adicionales:**

La instalación y mantenimiento de un servidor de aplicaciones conlleva costes financieros y temporales. Se debe considerar cuidadosamente si su implementación es necesaria, teniendo en cuenta los gastos asociados.

* **Posibles retrasos:**

Durante la implementación de un servidor de aplicaciones, pueden surgir retrasos, especialmente si la instalación se realiza detrás de un cortafuegos.

* **Programación exigente:**

Los requisitos de programación son más elevados para un servidor de aplicaciones, por lo que es recomendable contar con experiencia y conocimientos previos.

* **Ancho de banda:**

El uso simultáneo de varias aplicaciones extensas por parte de múltiples usuarios puede afectar negativamente la velocidad y el rendimiento general.

* **Impacto de problemas y errores:**

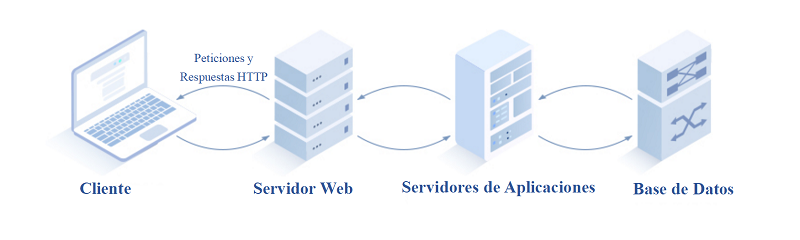
Aunque la agrupación es una ventaja del uso de servidores de aplicaciones, los problemas y fallos se vuelven más significativos. Dado que varios clientes dependen de una única fuente, los problemas con el software afectan a múltiples usuarios, y las soluciones a estos problemas suelen ser más complejas.

* **Cómo funciona un servidor de aplicaciones**

Gestión de solicitudes HTTP: Se encargan de procesar y manejar las solicitudes recibidas a través del protocolo HTTP o HTTPS, permitiendo la comunicación entre clientes (navegadores, aplicaciones) y el servidor. Manejo de sesiones: Facilitan la gestión de las sesiones de usuario, lo cual puede incluir la autenticación, seguimiento de estados y administración de cookies. Administración de recursos: Distribuyen recursos como memoria, conexiones a bases de datos, procesamiento y almacenamiento para asegurar una ejecución eficiente de las aplicaciones. Intermediación de protocolos y servicios: Actúan como intermediarios entre diferentes aplicaciones y servicios, traduciendo protocolos si es necesario y coordinando la comunicación entre distintos componentes dentro de una aplicación. Escalabilidad y balanceo de carga: Permiten distribuir equitativamente las solicitudes entre múltiples servidores para optimizar el rendimiento y evitar sobrecargar un único servidor. Seguridad: Proporcionan capas adicionales de seguridad como firewalls, filtrado de datos, cifrado y protección contra amenazas para garantizar la integridad y confidencialidad de los datos. Gestión del ciclo de vida de las aplicaciones: Ayudan en el despliegue, monitoreo, actualización y escalado adecuado.

* **¿Cómo funcionan juntos los servidores de aplicaciones y los servidores web?**

Los servidores de aplicaciones y los servidores web trabajan juntos para administrar las solicitudes de los clientes y ofrecer el contenido correcto al usuario. El servidor web siempre recibe primero una nueva solicitud. Si puede producir la información por sí mismo, lo hace y envía una respuesta HTTP. También comprueba que los datos que el usuario solicitó no estén ya en su caché.



Si el servidor web no puede acceder al contenido que el usuario necesita, reenvía la solicitud al servidor de aplicaciones. El servidor de aplicaciones procesa los datos y utiliza la lógica empresarial para proporcionar la información correcta. A continuación, devuelve la solicitud al servidor web, que la pasa al usuario. En ciertas arquitecturas, también puede configurar los servidores de aplicaciones para que gestionen las solicitudes HTTP por sí mismos.