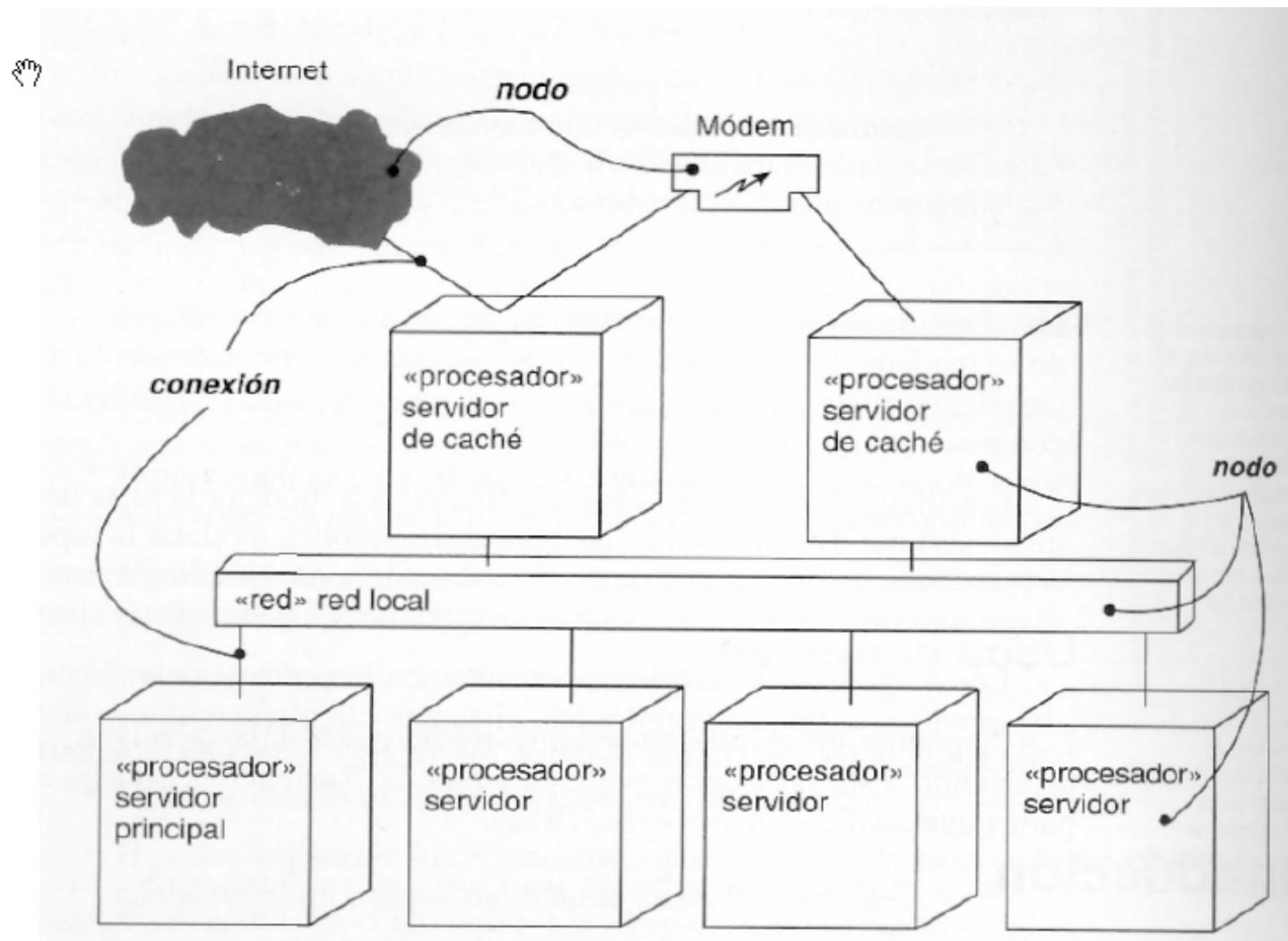
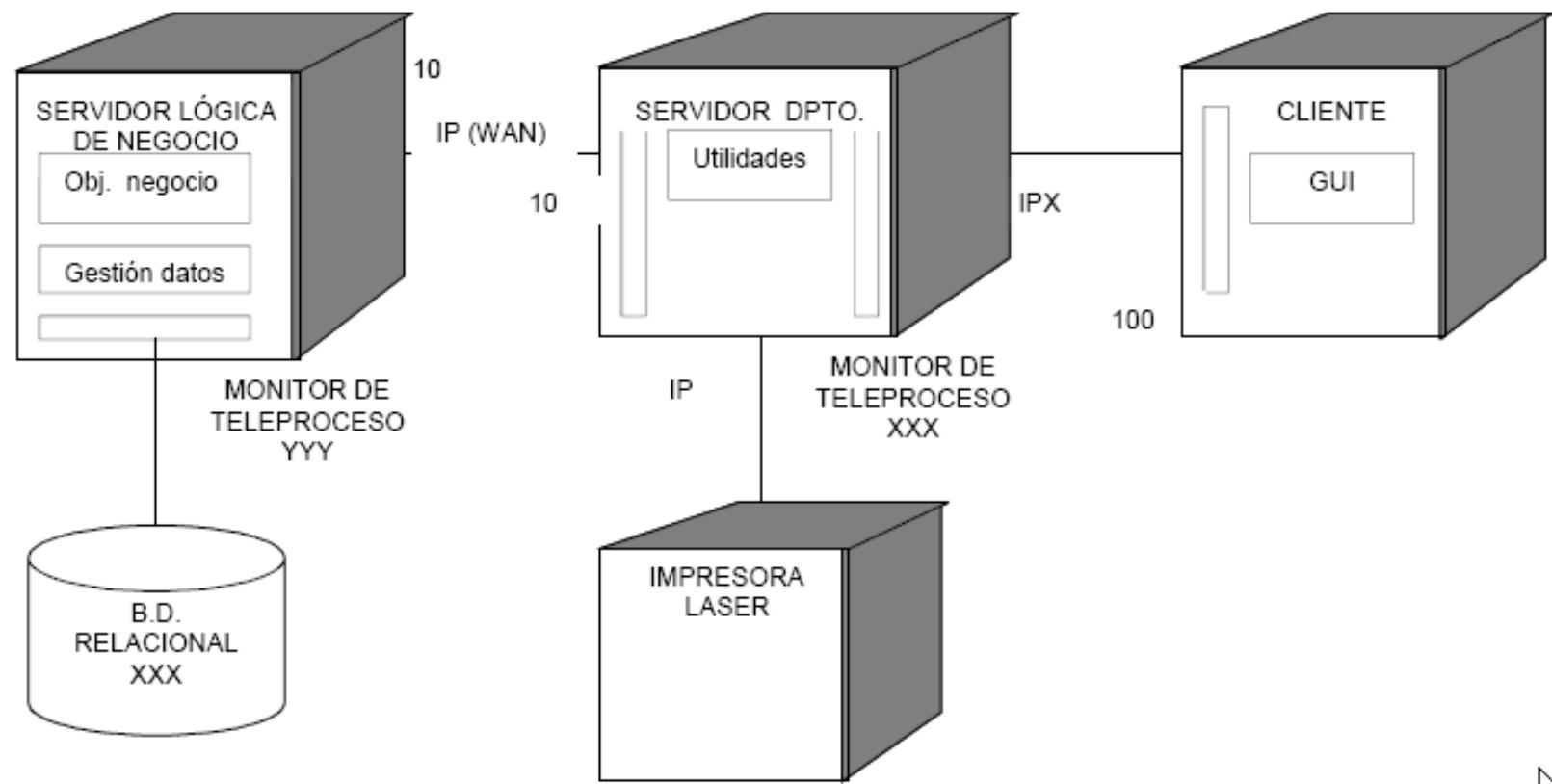


Diagrama de despliegue



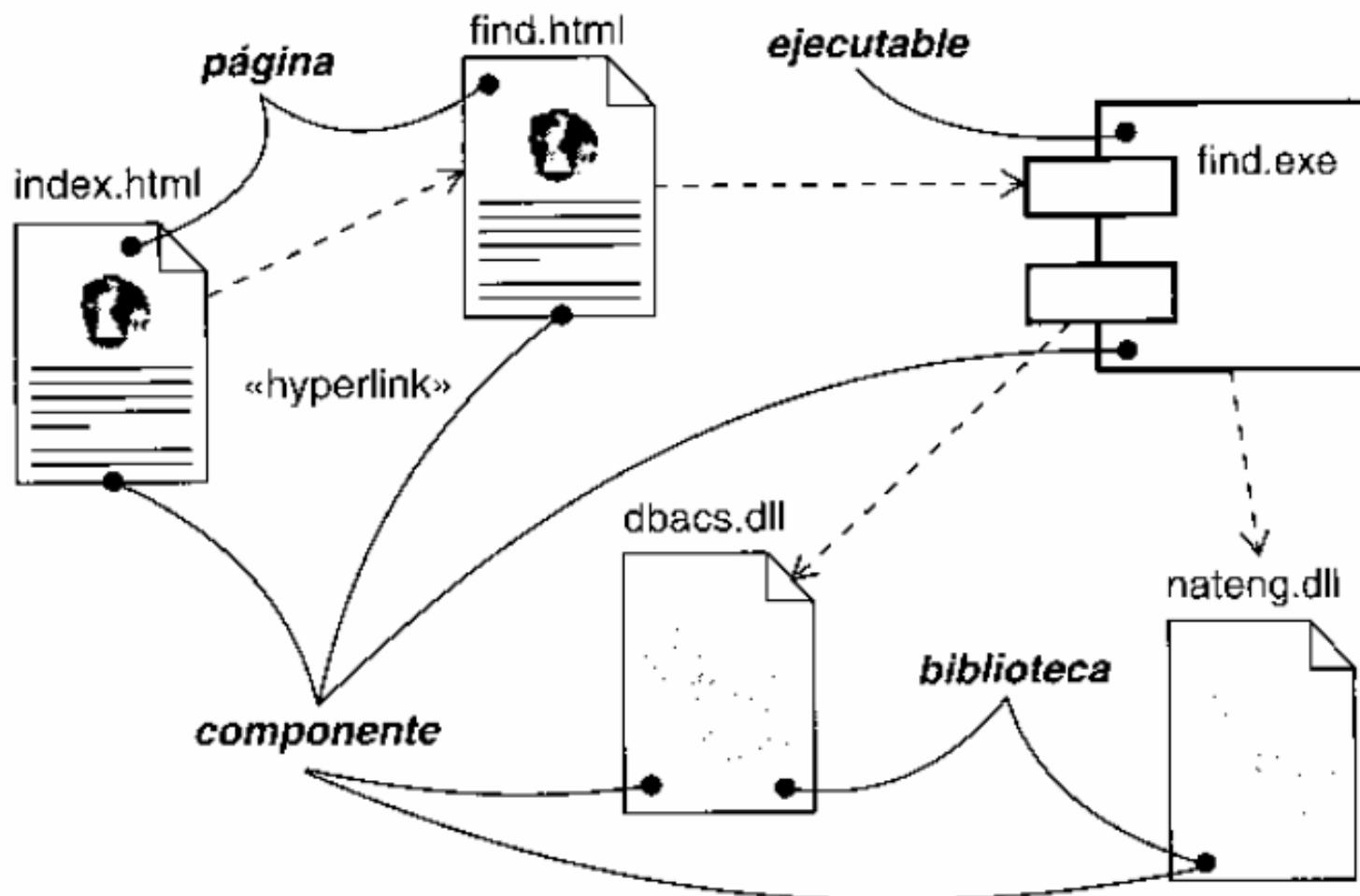
Se desea modelar la arquitectura de un sistema de altas prestaciones. El sistema tendrá un servidor principal que será respaldado por 3 servidores auxiliares. El acceso a la funcionalidad ofrecida por los servidores será ofrecido por dos servidores de caché que almacenarán peticiones ya satisfechas para no sobrecargar a los servidores. Los servidores de caché y el resto de servidores formarán todos parte de la misma red de área local. Uno de los dos servidores de caché estará conectado directamente a Internet. Los dos aceptaran peticiones a través de un módem compartido.

Diagrama de despliegue



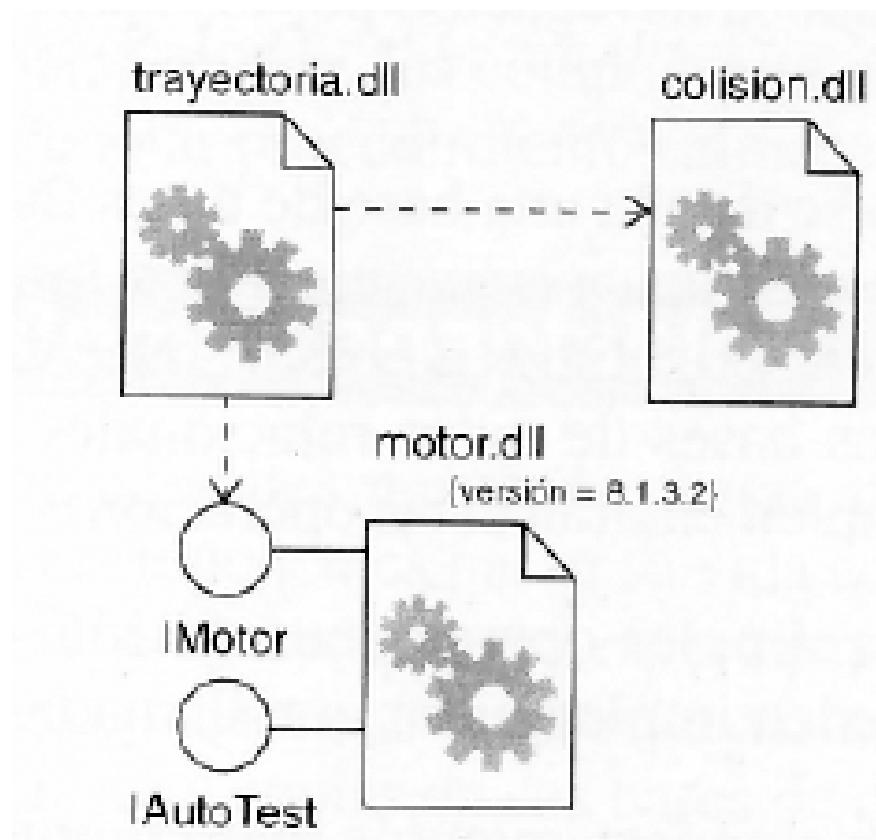
El diagrama representa una arquitectura compuesta por un servidor central de lógica de negocio y acceso a una base de datos relacional, en un monitor de teleproceso, al cual hay conectados 10 servidores departamentales, que proporcionan las utilidades del sistema, con clientes que contienen la interfaz gráfica(100) e impresora conectados a cada uno de ellos.

Diagrama de componentes



Se desea modelar parte de una aplicación web. La página de inicio del portal, llamado `index.html`, ofrece la posibilidad de realizar búsquedas. Para ello, se abre una nueva página llamada `find.html` en la que el usuario puede introducir la cadena de búsqueda. Al pulsar el botón de búsqueda, se invoca un ejecutable llamado `find.exe`. `find.exe` necesita para implementar la búsqueda hacer uso de dos bibliotecas de enlace dinámico llamadas `dbacs.dll` y `nateng.dll`.

Diagrama de componentes



Se desea modelar la parte de versión ejecutable de un robot autónomo centrada en las funciones de movimiento y cálculo del robot. La biblioteca de enlace dinámico **motor.dll** exporta una interfaz (**IMotor**) que a su vez es importada por el componente (**trayectoria.dll**). **motor.dll** exporta otra interfaz (**IAutoTest**) que será utilizada probablemente por otros componentes del sistema. Otro componente (**colision.dll**) proporciona funcionalidad que necesita **trayectoria.dll**.