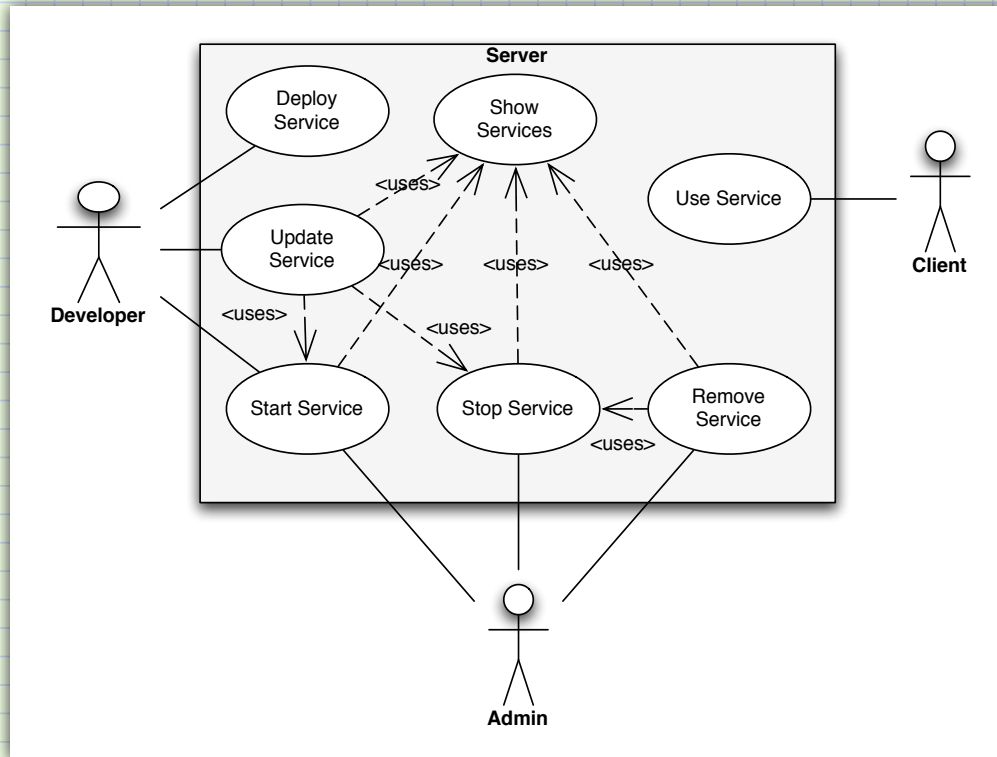


Use Cases

- A service is a function that resides on a server and is callable by a remote client. A service is independent of the mechanism by which it is invoked.



▼ Use a Service

- A client connects to a server, looks up a service by name, and calls one or more functions in the service. The functions may be blocking or non-blocking.

▼ Deploy a Service

- A developer creates a service and deploys it to a server that is already running. Deployment consists of transporting the service from the developer's workspace to the server, and making the server runtime aware of the service. The service may be started or removed at this point.

▼ Show Services

- An administrator connects to a server that is already running. A list of available services is shown to the admin.

▼ Start a Service

- The admin selects a service and clicks start. The server runtime starts the service and the service is now available for clients to invoke.

▼ Stop a Service

- The admin selects a service and clicks stop. The server runtime stops the service if no clients are currently using the service. The server runtime asks the admin if clients currently using the service may finish before stopping, or the service is stopped immediately. If the admin selects to allow clients to finish using the service, no new clients may be serviced.

▼ Update a service

- The developer selects a service and clicks update. The updated code in the developer's workspace is transported to the server. If the service is running, the service is stopped. The existing code is replaced with the new code. If the service was running, it is restarted.

▼ **Remove a service**

- The admin selects a service and clicks remove. The service is stopped if it is running. The service code is removed from the server.

▼ **Notes**

- The server must be available 24-7
- Services must be developed and packaged as an Eclipse plug-in
- The clients must know the difference between a service not found which is an error condition, and a service unavailable which would be handled as appropriate by the code calling the service. Developers could choose to drop the service request, retry the service request immediately, or at a later time.
- Services must be callable by SOAP allowing us to interface our legacy perl scripts
- Services must be easily called by an Eclipse client. Our current solution uses RMI.