

Założenia

Framework do tworzenia własnego serwera (serwer bazowy obsługujący wiele protokołów).

Istnieje w nim możliwość zmiany i wyboru różnych protokołów. Dołożenie obsługi kolejnego protokołu ma być łatwa, prosta i przyjemna.

Opis

Serwer

Aplikacja serwera działa jako daemon. Dzięki temu wszelkie instancje serwisów mogą być wywoływane w dowolnym momencie „na żądanie”.

Głównym elementem serwera jest *Server IO Module (IOM)*, w którego skład wchodzi również:

- *Basic Loader Controller (BLC)* – obsługujący podstawowe żądania (tj. start, zatrzymanie, restart, przeładowanie serwera i serwisów)
- *Configuration Repository (CR)* – czyli główna konfiguracja serwera
- *Exeptions Controller (Ex)* – metody do obsługi wszelkich wyjątków

Uzupełnieniem tego jest *Services Repository (SR)* – baza konfiguracji poszczególnych serwisów (usług).

Centralną jednostką zarządzającą serwerem jest *Server Manager (SrMan)*.

Dodatkowe funkcjonalności zawiera się biblioteka narzędzi serwerowych (*Server Tools Module – STM*) czyli podserwery logów, statystyk, raportów i historii.

Serwis

Serwis jest powoływany do życia przez BLS za pomocą SR.

Centralną jednostką zarządzającą serwisem jest *Service Manager (ScMan)*. Natomiast głównym elementem przetwarzającym wiadomości jest *Application Controller (App)*.

Dane wpływające do serwera przechodzą przez *Incoming Data Module (IDM)*, gdzie są poddawane podstawowej obróbce, filtrowaniu i konwersji na wewnętrzny system reprezentacji. Tak gotowe paczki odbiera ScMan, i przekazuje do App, który przetwarza dane zgodnie z obsługą zaprogramowanego protokołu. Po przetworzeniu dane są przesyłane dalej do *Outgoing Data Module (ODM)*. Tam dane są odpowiednio przetwarzane (np. szyfrowanie, porcjowane) i przesyłane na wyjście.

Ścieżka pracy:

- bity z medium
- parsowanie na poziomie syntaktycznym
- na wyjściu jest obiekt
- kolejka wiadomości
- utworzenie wiadomości zwrotnej
- operacja odwrotna do parsowania
- wyjście na adapter
- bity do medium

Wszelkie komunikaty i żądania wewnętrzne są przesyłane z IDM i ODM do ScMan systemowym kanałem danych. Informacje te są zapisane w formie takiej samej jak dane

przetwarzane przez serwis, jednakże posiadają dodatkową flagę specjalną typu *internal message*.

Zarządzanie sesją i obsługa bibliotek narzędzi jest realizowana przez ScMan.

Schemat

Server Application

Main application is daemon, which can be connect via console or web admin (service).

- Server IO Module
 - Server Manager
 - Basic Loader Controller (startup, shutdown, reload, flush)
 - Configuration Repository
 - Exeptions Controller
- Services Repository
 - This is repository of configuration, filters and data standards for each services protocols.
 - Custom Server Component
- Server Tools Module
 - Logs Server Manager
 - Statistics Server Manager
 - Reports Server Manager
 - History Server Manager
 - Clients (connections) repository

Service Application

- Service Module
 - Service Manager
 - Application Controller
- Incoming Data Module (synonyms: Client Module or Transformer Module)
 - Prerouting Controller - automat - flow controller (protocol transformer, parser, object)
 - Filters Controller
- Outgoing Data Module (synonyms: Application Module or Responser Module)
 - Automat - flow controller (object, revert parser, data response)
 - Message Mapper → (delivers) Action → (changes) State → (uses) MM
 - external data access (horizontal connection via services - optional)
- Session Module
 - Session Controller
- Service Tools Module
 - Logs Client Controller
 - Statistics Client Controller

- Reports Client Controller
- History Client Controller
- Custom Client Controller

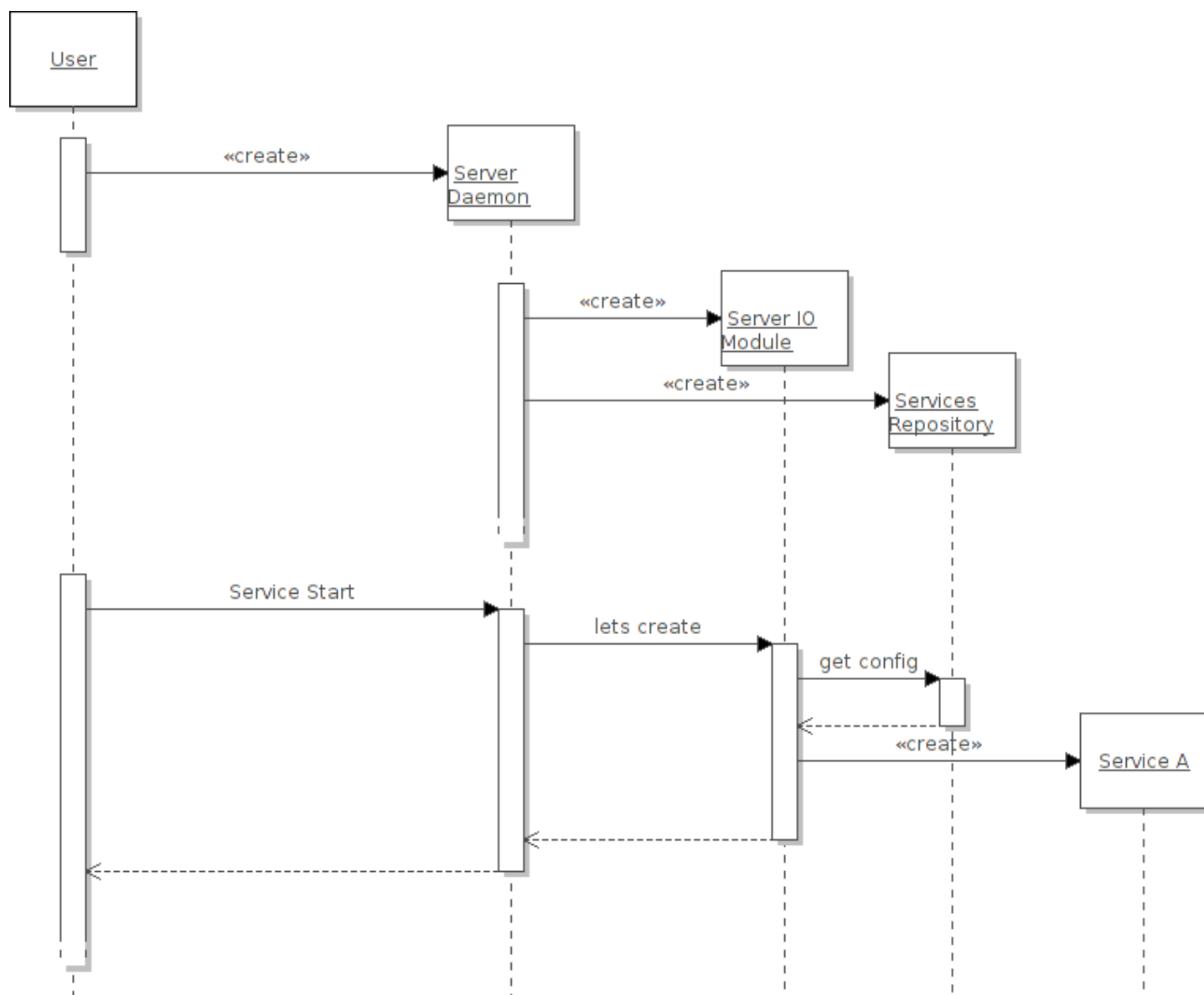
Diagram sekwencji tworzenia serwisu

Diagram komponentów

