

## Gem #101: Сервера SOAP/WSDL

Автор: Pascal Obry, EDF R&D

Краткое содержание: В данном Gem описывается построение сервера для обеспечения веб-услуг в сети.

### Давайте начнём...

Это первая часть Gem из двух частей на SOAP (Simple Object Access Protocol).

В этом Gem #101 мы будем строить сервер SOAP, и вы увидите, что с Ada это довольно просто!

Давайте возьмём простую спецификацию пакета, такую как:

```
package Temperatures is

  type Celsius is new Float;
  type Fahrenheit is new Float;

  function To_Fahrenheit (C : Celsius) return Fahrenheit;
  function To_Celsius    (F : Fahrenheit) return Celsius;

end Temperatures;
```

Тело пакета не показано здесь, но это часть исходных пакетов, которые могут быть загружены (см. ниже).

Первым шагом является создание WSDL (языка описания веб-службы). WSDL – это язык XML для описания веб-служб. В WSDL мы находим описание типов и спецификаций подпрограмм. WSDL похож на IDL, но основан на XML.

Для создания WSDL AWS поставляется с инструментом ADA2WSDL на основе ASIS:

```
$ ada2wsdl temperatures.ads -a http://localhost:8888 -o temperatures.wsdl
```

Опции:

```
-a http://...           Specifies the end-point for the Web services.
                        Определяет конечную точку для Веб-сервисов.

-o temperatures.wsdl    Outputs WSDL into temperatures.wsdl.
                        Выводит WSDL в temperature.wsdl.
```

Из этого WSDL можно создавать заглушки (для вызова веб-служб) или скелеты (для реализации веб-служб). В этой первой части мы создаём сервер, поэтому нам не нужны заглушки. AWS поставляется со вторым инструментом под названием wsdl2java для генерации всего необходимого кода:

```
$ wsdl2aws -nostub -cb -spec temperatures -main soap_server temperatures.wsdl
```

Опции:

```
-spec temperatures     To use the routines as implemented in Temperatures
                        unit.
                        Использовать подпрограммы, как реализовано в модуле
                        Температур.

-cb                    Generates the SOAP callbacks using the routines
```

found in the spec specified above.

Генерирует обратные вызовы SOAP, используя подпрограммы, найденные в спецификации, определенной выше.

```
-main soap_server      Generates a main named soap_server, this main
                        program starts the SOAP server by referencing a SOAP
                        dispatcher using the callback routines.
                        Генерирует главный именованный soap_server, эта основная
                        программа запускает SOAP-сервер, ссылаясь на диспетчера
                        SOAP, используя процедуры обратного вызова.
```

Используя эти три опции очень удобно для создания сервера, который обеспечивает Веб-сервисы и ничто больше. Последние действия должны только скомпилировать сервер и выполнить его:

```
$ gnatmake -gnat05 -Psoap_server
$ ./server
```

На этом этапе сервисы доступны в сети и могут быть вызваны другими программами, возможно построенными на других языках (Java и C# являются наиболее распространёнными).

Во второй части этой серии мы увидим, как вызывать эти сервисы из Ada с помощью AWS.

### Связанный со статьёй текст программы

#### [soap\\_server.zip](#)

```
soap_server.gpr
```

```
with "aws";
project SOAP_Server is
  for Source_Dirs use (".");
  for Main use ("soap_server.adb");
end SOAP_Server;
```

```
temperatures.ads
```

```
1. package Temperatures is
2.
3.   type Celsius is new Float;
4.   type Fahrenheit is new Float;
5.
6.   function To_Fahrenheit (C : Celsius) return Fahrenheit;
7.   function To_Celsius   (F : Fahrenheit) return Celsius;
8.
9. end Temperatures;
```

```
temperatures.adb
```

```
.1.
2. package body Temperatures is
3.
4.   function To_Fahrenheit (C : Celsius) return Fahrenheit is
5.   begin
6.     return Fahrenheit (Float (C) * 9.0 / 5.0 + 32.0);
7.   end To_Fahrenheit;
8.
9.   function To_Celsius   (F : Fahrenheit) return Celsius is
10.  begin
11.    return Celsius ((Float (F) - 32.0) * 5.0 / 9.0);
```

```
12.     end To_Celsius;  
13.  
14. end Temperatures;
```

Файлы примеров Ada Gems распространяются AdaCore и могут быть использованы или изменены для любых целей без ограничений.

### **Об авторе**

Сведений нет.

*Last Updated: 10/13/2017*

*Posted on: 3/14/2011*

### **Обсуждение...**