

# Memoria P2 - SUN RPC

Salvador Romero Cortés

---

## SUN RPC

Para esta primera práctica usamos SUN RPC con el lenguaje de programación C. Primero definimos una serie de structs para almacenar los datos de los parámetros de entrada de cada procedimiento:

```
struct pareja_double{
    double num1;
    double num2;
};

struct pareja_int{
    int num1;
    int num2;
};

struct vec3{
    double x;
    double y;
    double z;
};

struct pareja_vec3{
    vec3 v1;
    vec3 v2;
};

struct vec2{
    double x;
    double y;
};

struct pareja_vec2{
    vec2 v1;
    vec2 v2;
};

struct vec2escalar{
    vec2 v;
    double escalar;
};

struct vec3escalar{
    vec3 v;
    double escalar;
};
```

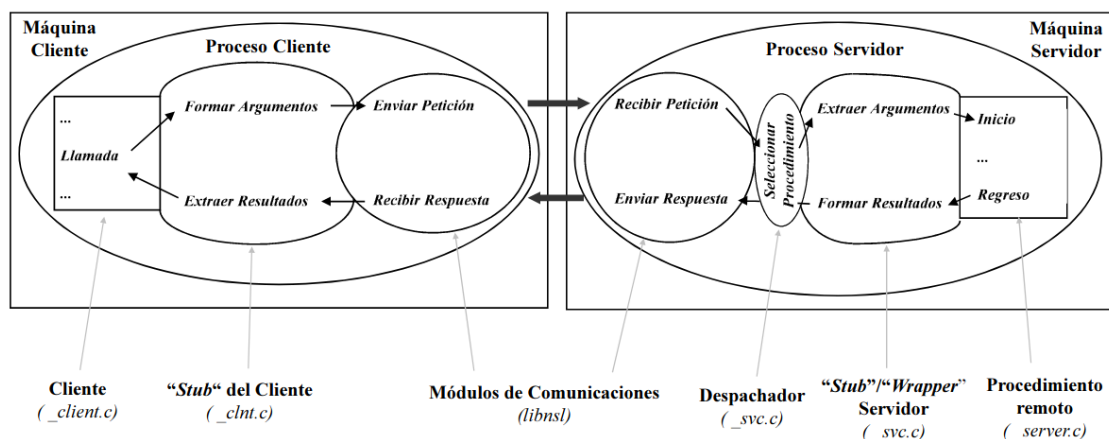
De esta forma podemos ajustarnos al requisito de RPC de tener un único parámetro de entrada. Una vez creados los structs, definimos las operaciones de la calculadora:

```

program CALCULADORA {
    version CALCVER {
        double SUMAR(pareja_double) = 1;
        double RESTAR(pareja_double) = 2;
        double MULTIPLICAR(pareja_double) = 3;
        double DIVIDIR (pareja_double) = 4;
        int DIVISIONENTERA(pareja_int) = 5;
        int MODULO (pareja_int) = 6;
        vec3 SUMAVEC3(pareja_vec3) = 7;
        vec2 SUMAVEC2(pareja_vec2) = 8;
        vec3 RESTAVEC3(pareja_vec3) = 9;
        vec2 RESTAVEC2(pareja_vec2) = 10;
        vec2 PESCALAR2(vec2escalar) = 11;
        vec3 PESCALAR3(vec3escalar) = 12;
    } =1;
} = 0x20000155;

```

Ahora usamos la orden `rpcgen -Nca calculadora.x` para generar los ficheros necesarios para programar el sistema. Los ficheros que se generan incluyen los ficheros de cliente servidor así como sus stubs necesarios para funcionar. Un diagrama de como funciona es:



Una vez tenemos los ficheros generados ya consiste en implementar las funciones en el servidor de la calculadora y hacer llamadas a él desde el cliente.

## Funcionamiento del programa

Para ejecutar el programa ejecutamos `rpcgen`, luego compilamos con `make -f Makefile.calculadora` y sustituimos el `calculadora_client.c` y el `calculadora_server.c` por los que se incluyen en la práctica (rpcgen los sustituye por unos en blanco, por eso es necesario copiarlos despues de hacer el rpcgen).

Una vez compilado ejecutamos primero el servidor con `./calculadora_server` y desde otra terminal lanzamos el `./calculadora_client`.

Para lanzar el cliente es necesario pasarle una serie de argumentos. Para ello es necesario saber primero si vamos a trabajar con operaciones más sencillas (suma, resta, producto...) o si vamos a trabajar con vectores. En caso de que sean operaciones sencillas pondremos

```
./calculadora_client localhost num1 operador num2
```

Donde localhost es el servidor donde esté el calculadora\_server, num1 y num2 son los números que utilizamos en el cálculo y operador es un carácter que indica que operación se va a ejecutar. Los valores posibles del operador son:

```
+ para la suma
- para la resta
x para el producto
/ para la division
e para la division entera
% para el módulo
```

Una vez ejecutado se nos devolverá el resultado y el servidor indicará que operación está realizando. Por ejemplo:

<pre>&gt; ./calculadora_client localhost 12.32 x 3.1 Resultado de 12.320000 x 3.100000 = 38.192000 &gt; █</pre>	<pre>&gt; ./calculadora_server Haciendo producto de 12.320000 con 3.100000 █</pre>
---	--

Se incluyen también una serie de medidas para que no se pueda dividir por 0.

<pre>&gt; ./calculadora_client localhost 12 / 0 Resultado de 12.000000 / 0.000000 = 0.000000 &gt; █</pre>	<pre>&gt; ./calculadora_server No se puede dividir por 0 █</pre>
---	--

Para acceder al modo de operaciones con vectores lanzaremos el cliente con:

```
./calculadora_client localhost -a
```

De esta forma se nos presentará un menú interactivo donde escoger la operación con vectores que queramos usar. Están disponibles:

1. Suma de vectores
2. Resta de vectores
3. Producto escalar

Para cada operación con vectores se nos preguntará si queremos vectores de dimensión 2 o 3. Una vez elegido, introduciremos los valores de los vectores y nos devolverá el resultado.

<pre>&gt; ./calculadora_client localhost -a Bienvenido al modo avanzado Selecciona una operacion: 1. Suma de vectores 2. Resta de vectores 3. Producto escalar 1 Selecciona la dimension del vector (2 o 3): 3 Leyendo vector 1 Introduce las 3 componentes del vector: (separadas por un salto de linea) 10.2 11.2 13.1 Leyendo vector 2 Introduce las 3 componentes del vector: (separadas por un salto de linea) 4 5 6 Resultado de la operacion con vectores: {14.200000, 16.200000, 19.100000}</pre>	<pre>&gt; ./calculadora_server Sumando vectores █</pre>
---	---

