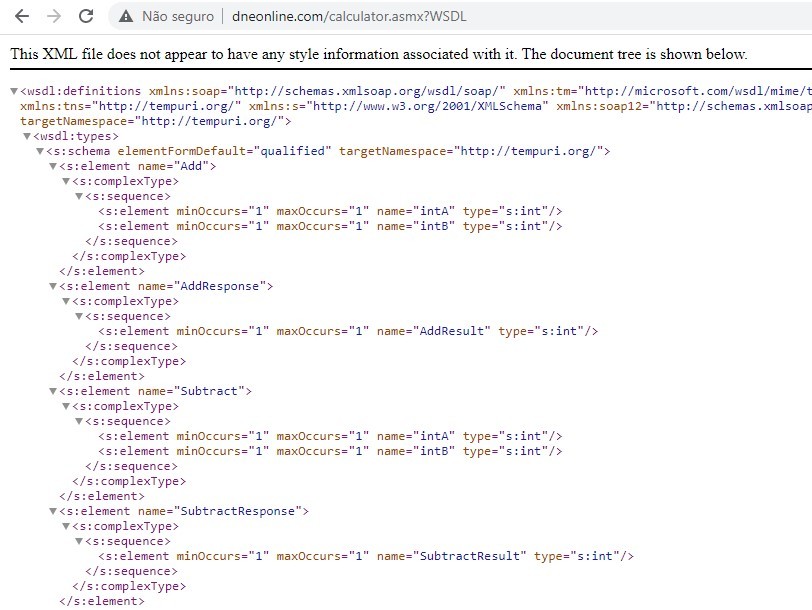
Você precisa chamar um **webservice**, mas descobre que não pode usar **PI**e agora?

Quando eu estava começando com **ABAP**me deparei com uma situação dessas, na época tentei de várias maneiras e não consegui, até que a demanda acabou indo pra outra pessoa que disse saber fazer, ao final do projeto perguntei como havia sido feito, mas não tive resposta, olhei no código e não fazia muito sentido.

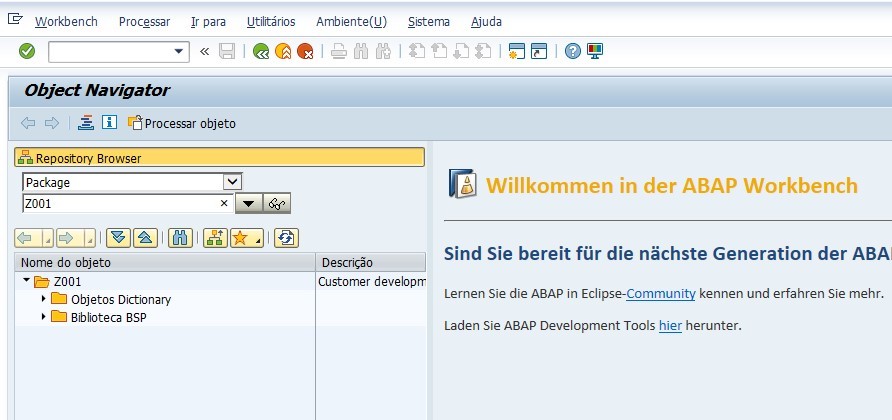
Dedico esse artigo ao **Diego**do **passado**, que queria demais ter um **passo-a-passo** desses.

**Resolução**

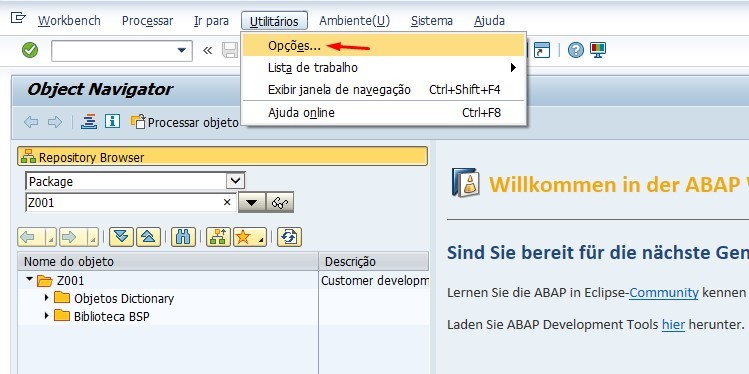
Primeiro precisamos ter o **WSDL**do webservice que iremos chamar, para esse nosso tutorial vou usar o de uma calculadora que é bastante famoso e gratuito. (http://www.dneonline.com/calculator.asmx?WSDL)



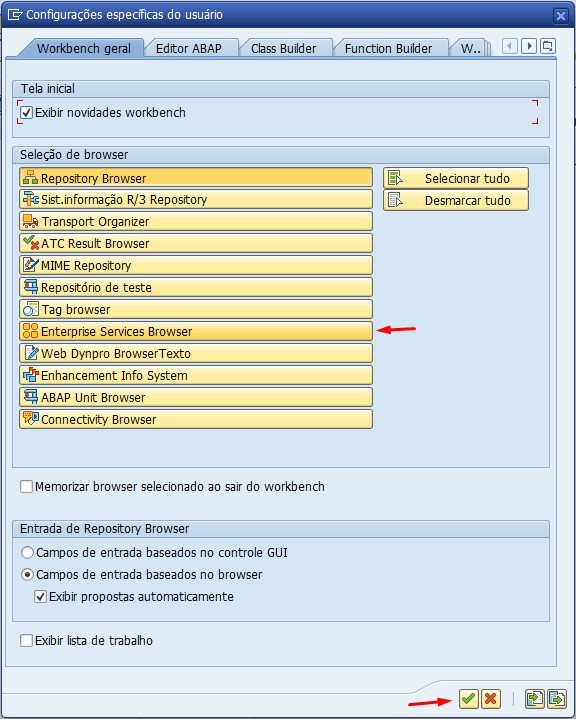
Um **WSDL**é um arquivo ou uma **URL**que contém a definição do **serviço**que iremos chamar. Em posse desse link acessamos a **SE80**.



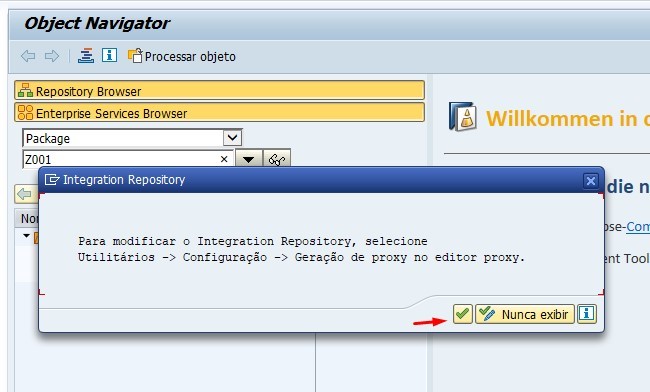
Caso não tenha a opção de **Enterprise Services Browser** você pode habilitar ela através do menu **Utilitários -> Opções**.



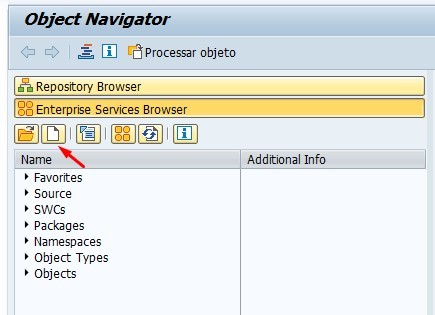
Marcar o **Enterprise Services Browser** e clicar em OK.



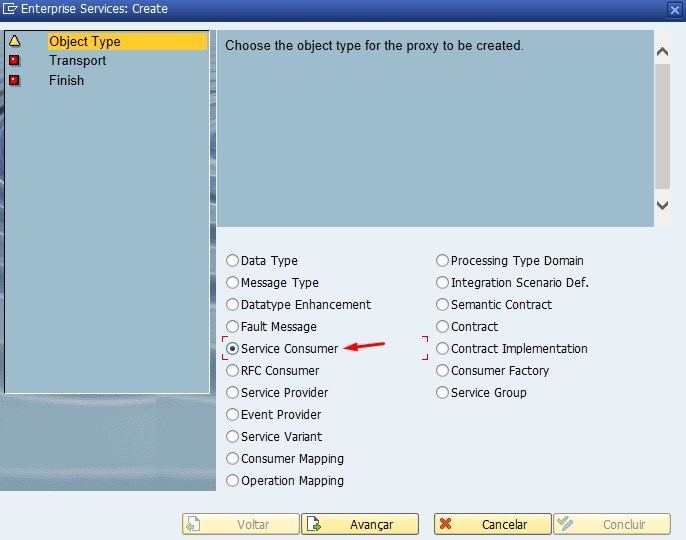
E então irá aparecer o botão do **ESB**, clique nele.



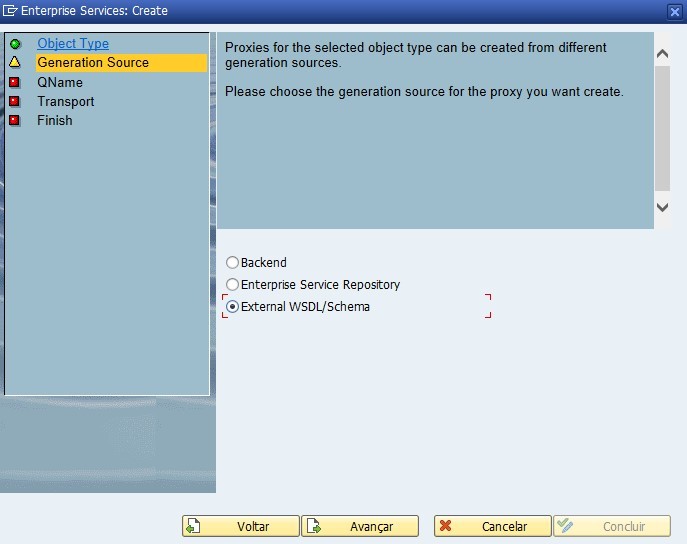
Caso apareça o **popup** conforme acima, clique em **ok**. Após selecionar o ESB, clique em **Novo**.



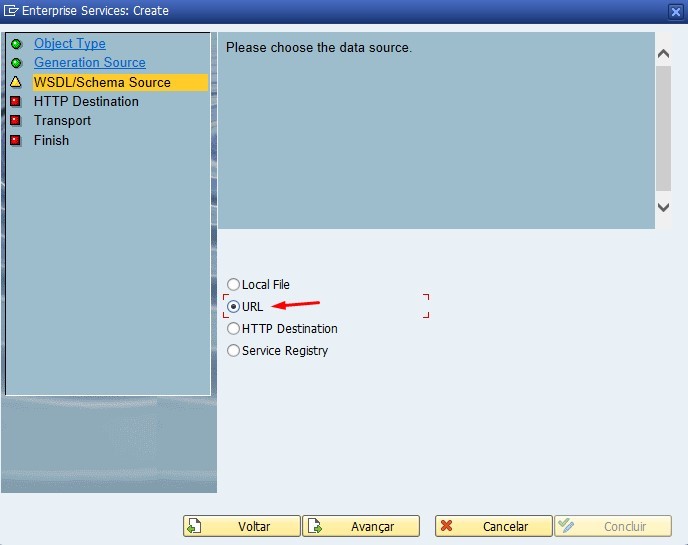
Selecione **Service Consumer**.



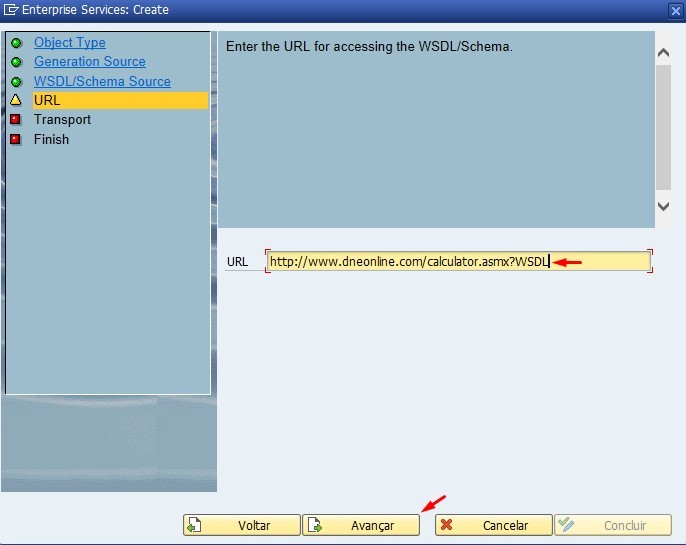
Clique em Avançar e escolha '**External WSDL/Schema**'.



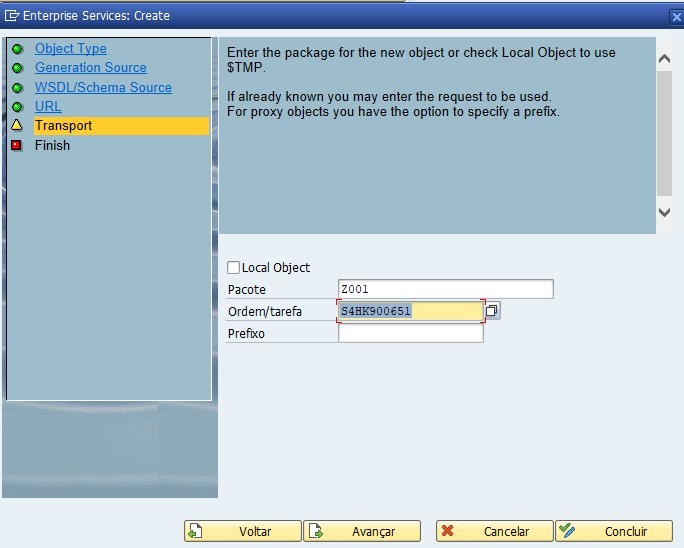
Clique em Avançar e escolha '**URL**' (Caso tenha o arquivo salvo pode também fazer via '**Local File**').



Clique em Avançar e preencha com a **URL**do **WSDL**.



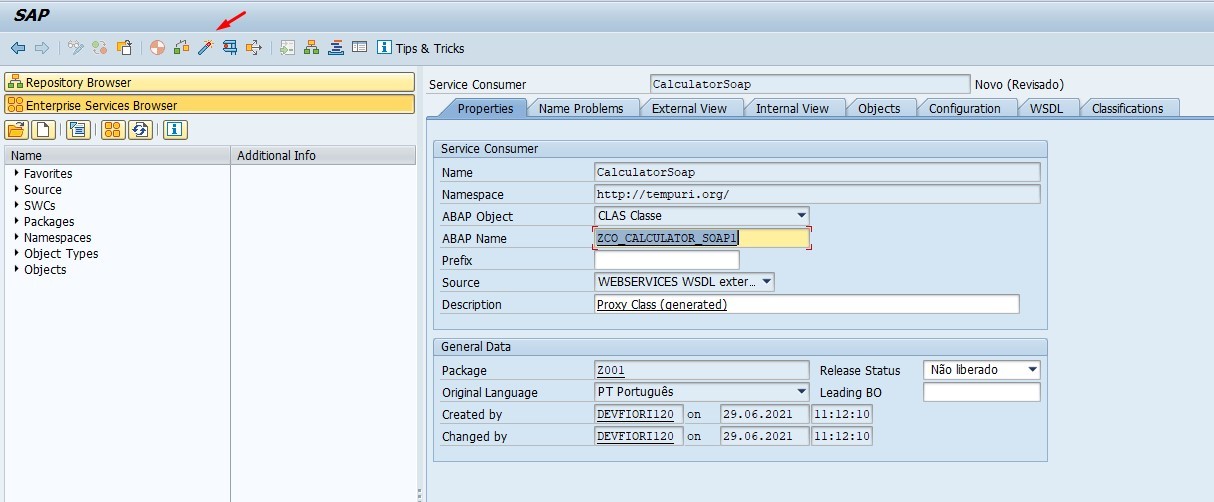
Clique em Avançar e informe **Pacote**e **Request**.



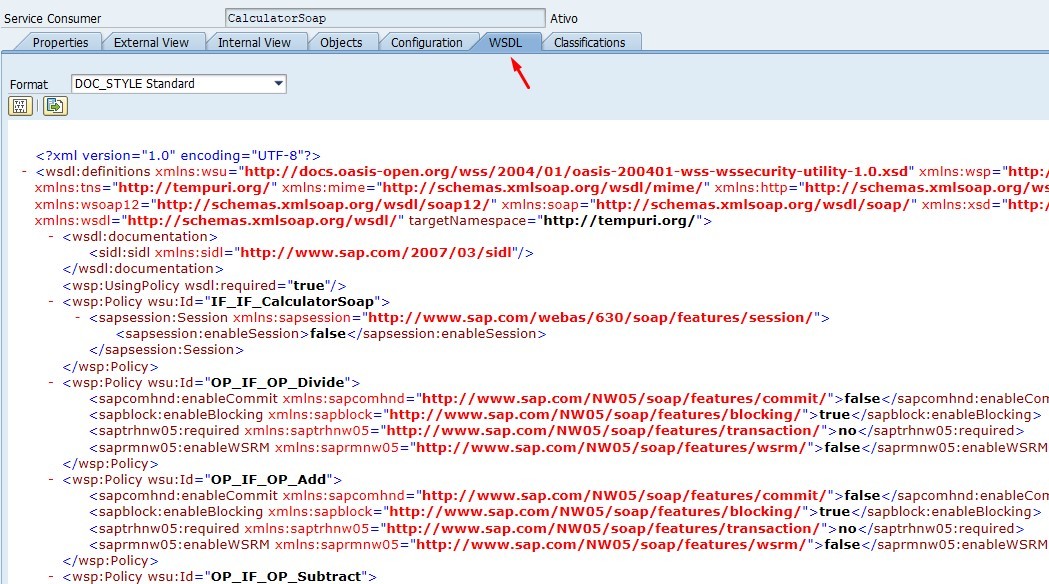
Pode aparecer um **popup** se não inserir prefixo, **não**é obrigatório, basta dar **enter**(ou clicar no botão de ok).



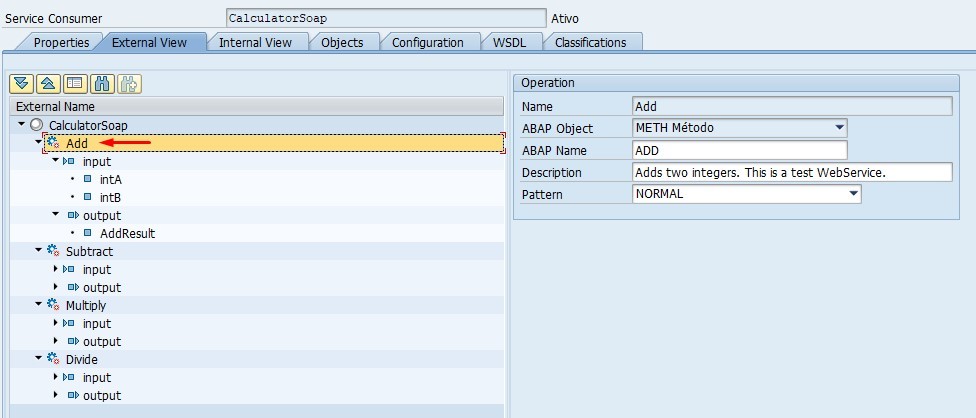
Então aparecerá a tela com o **Service Consumer** criado, basta ativar. (Guarde o **ABAP Name** do **Service Consumer** criado, pois usaremos mais pra frente).



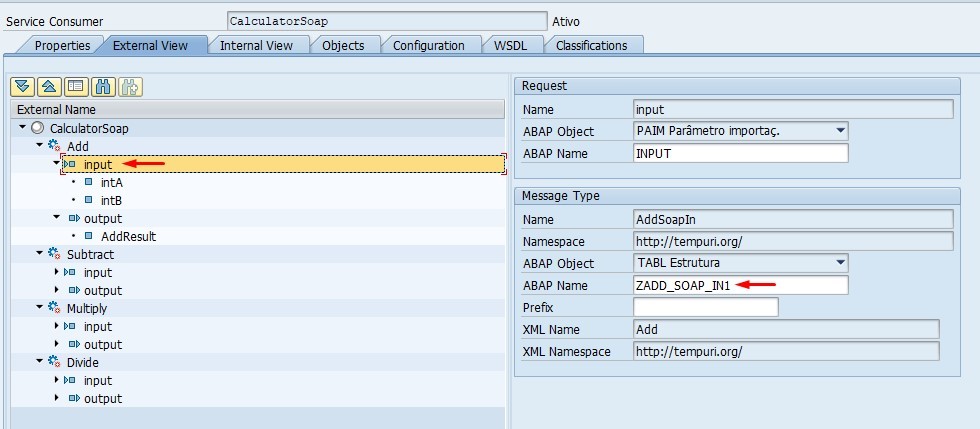
Se olharmos na aba **WSDL**veremos que ele puxou automaticamente da **URL**a definição inteira do **WebService** que iremos chamar.



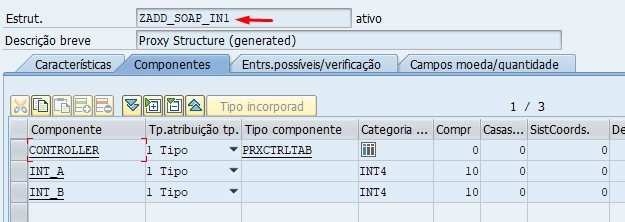
Se olharmos na aba **External View** conseguimos ver todos os métodos do **WebService**que podemos chamar, nesse nosso exemplo usaremos o método da soma (**Add**).



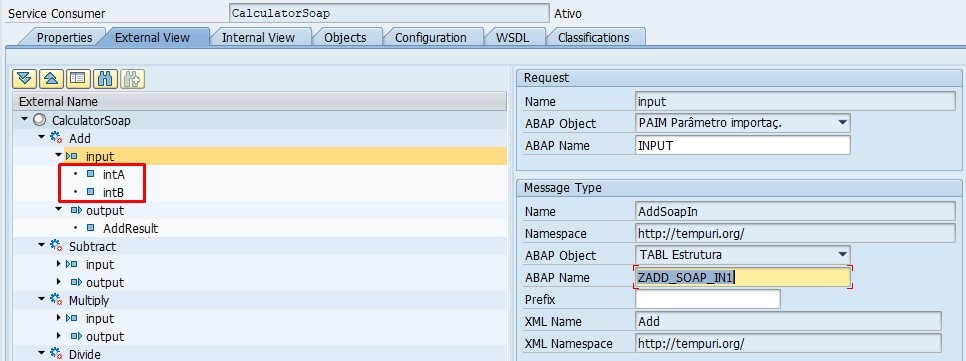
Podemos ver que ele já cria os tipos de entrada e saída do **WebService**, se clicarmos em **input**, ele abre as informações na tela da direita e podemos dar um duplo clique no nome ABAP (**ZADD\_SOAP\_IN1**no nosso exemplo).



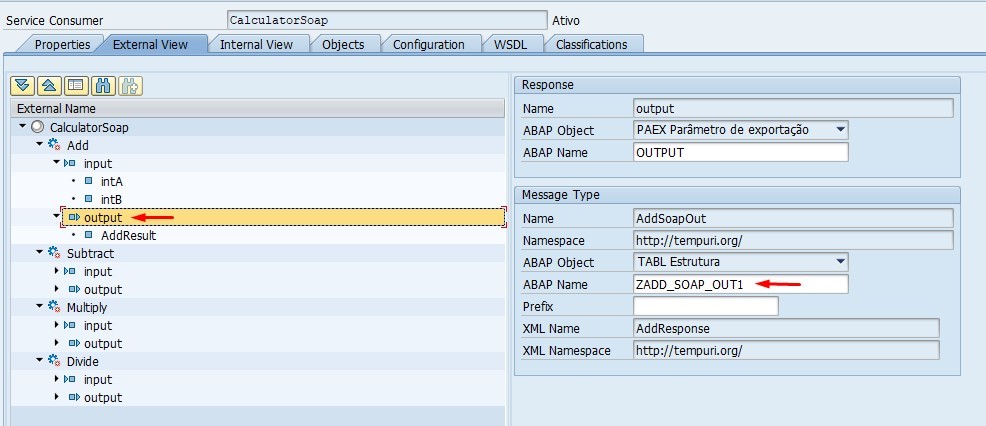
Ao dar o duplo clique no nome vemos que ele gerou uma estrutura na **SE11**com os campos e os tipos corretos já, essa **estrutura**que vamos declarar no nosso programa **ABAP**para preencher com os dados que enviaremos ao **WebService**.



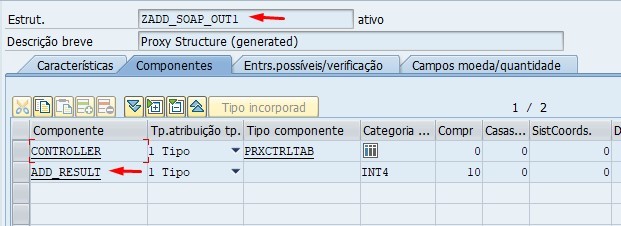
Vemos que a estrutura criada possui os mesmos campos, inclusive com o mesmo nome que está declarado no **WSDL**.



E a resposta que teremos será recebida através da estrutura de **output**, que foi gerada automaticamente também (**ZADD\_SOAP\_OUT1**).

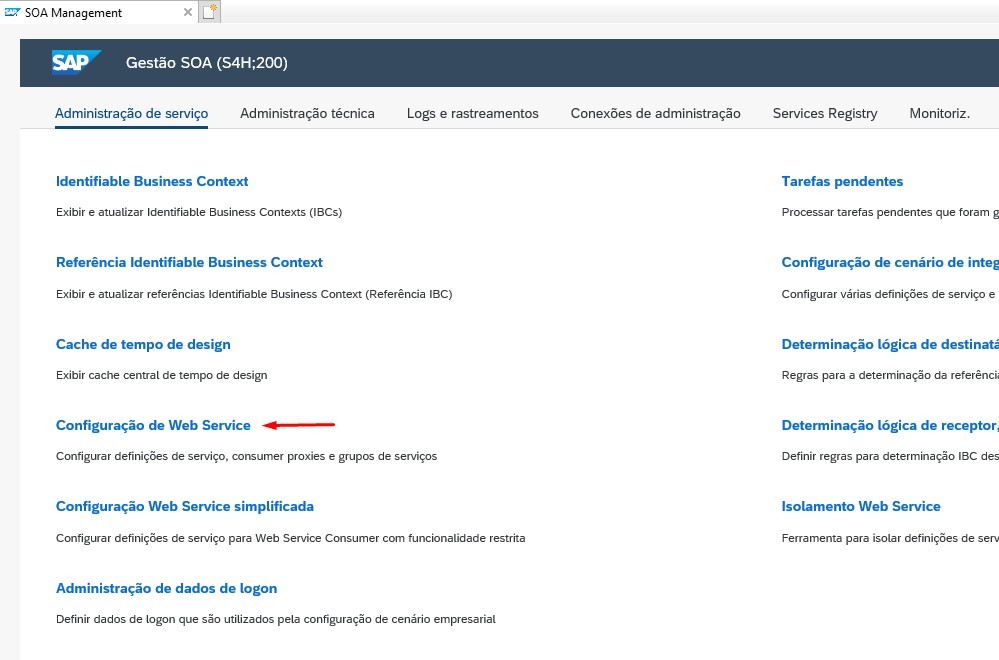


Essa estrutura criada na **SE11**que iremos declarar e usar no nosso código **ABAP** para ler o retorno do **WebService**.

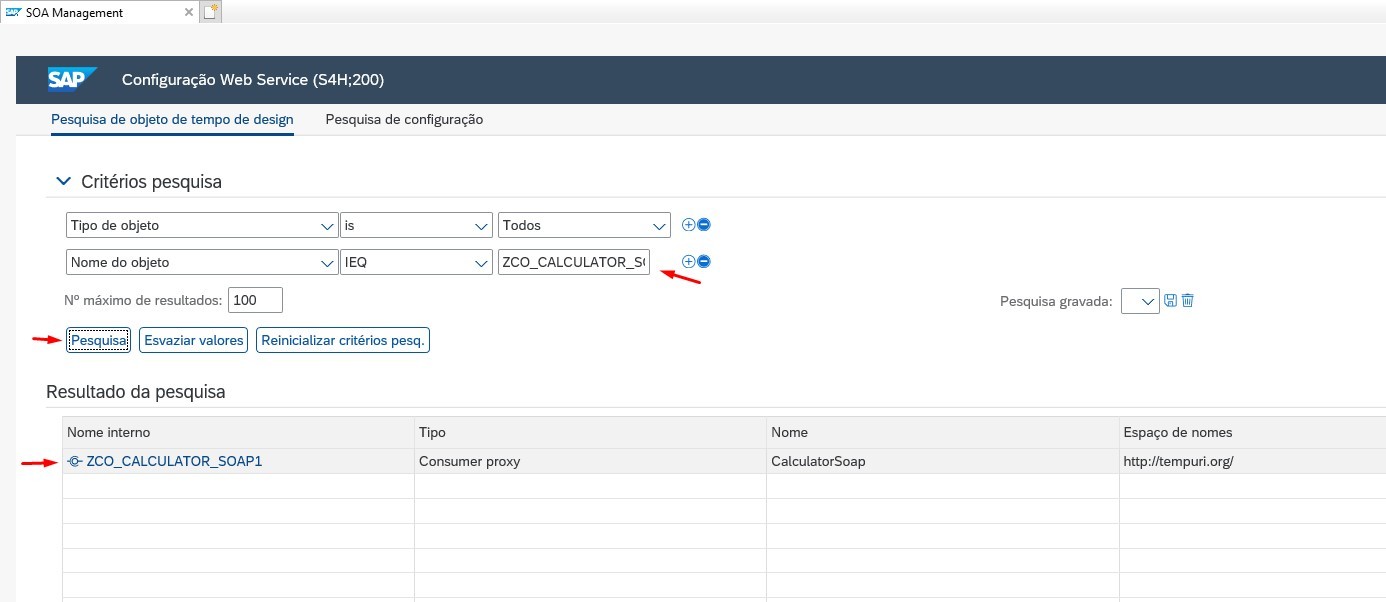


Agora que já criamos e entendemos o **Service Consumer**, vamos criar a nossa porta lógica na transação **SOAMANAGER**.

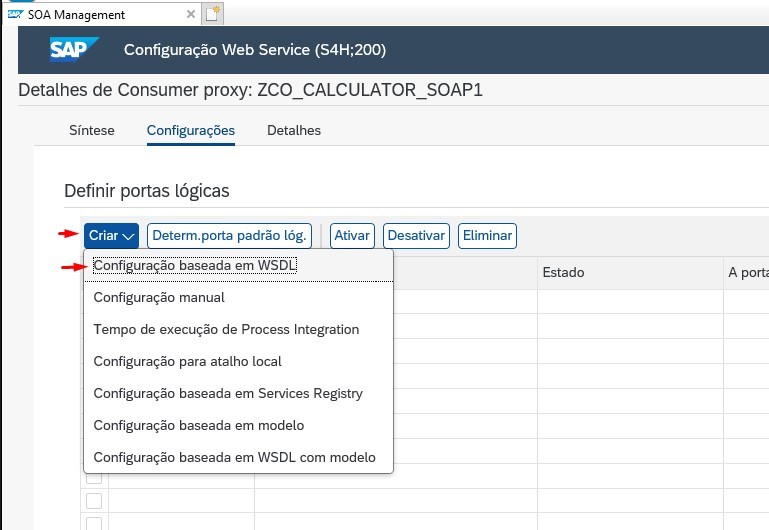
Ao executar a transação ele irá abrir o **navegador**da internet (IE, Edge, Firefox, Chrome, Opera) na tela abaixo, clique em '**Configuração de Web Service**'.



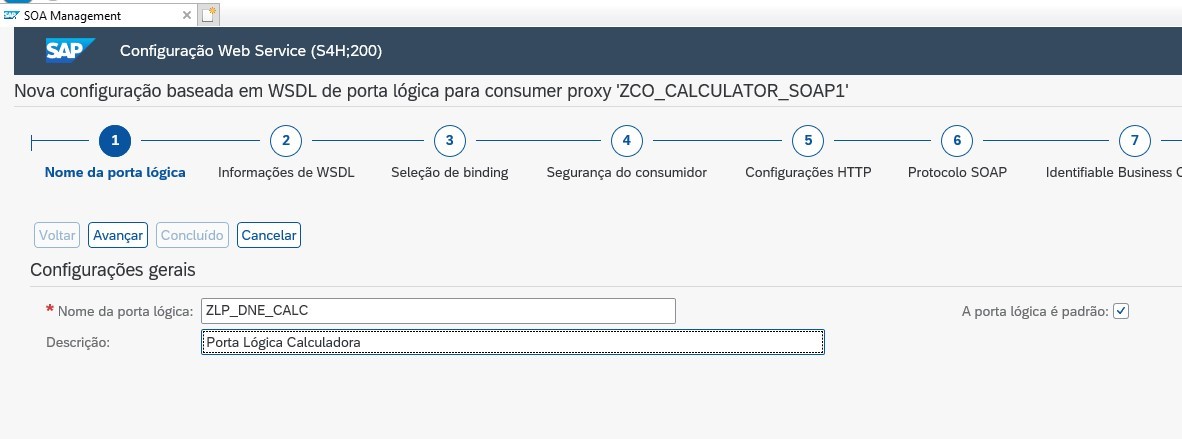
Lembra do **ABAP Name** do **Service Consumer** que anotamos? É pra usar agora, coloque o valor no filtro de '**Nome do Objeto**' e clique em **Pesquisa**. Quando aparecer o nome clique no nome. (Se não aparecer é porque faltou ativar o objeto no SAP).



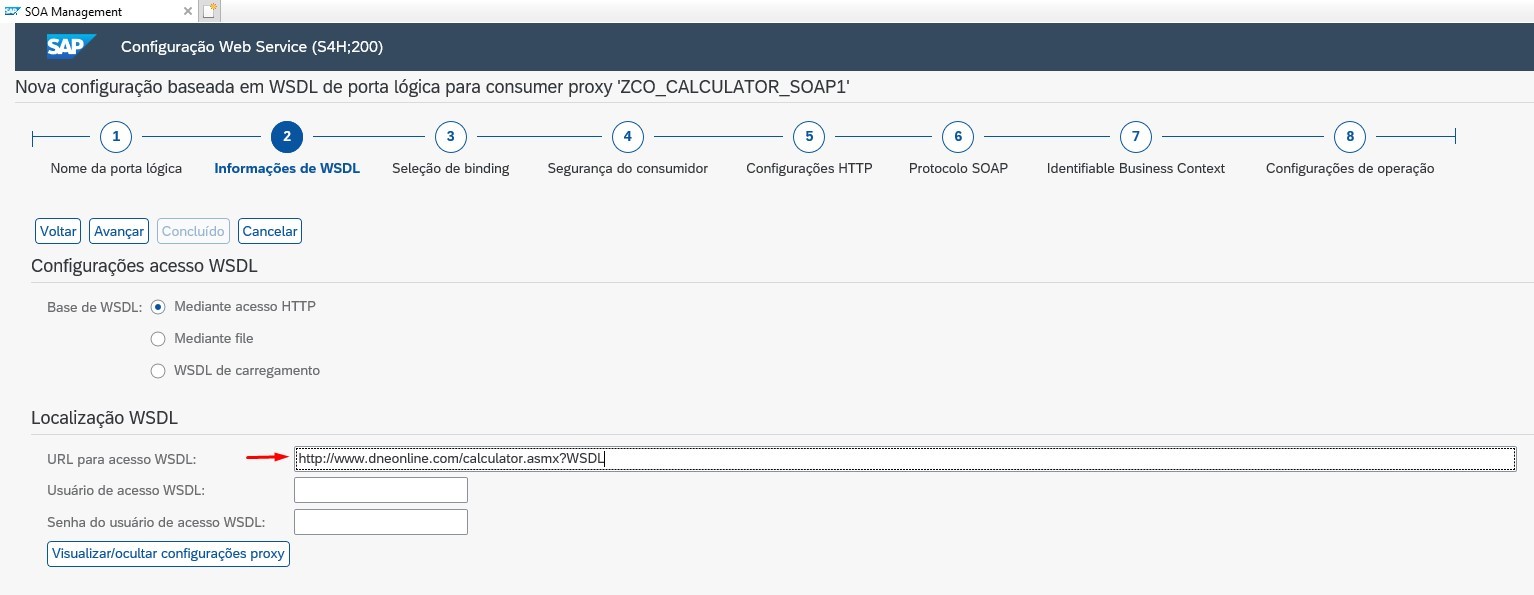
Na tela que irá abrir clique em **Criar->Configuração baseada em WSDL**.



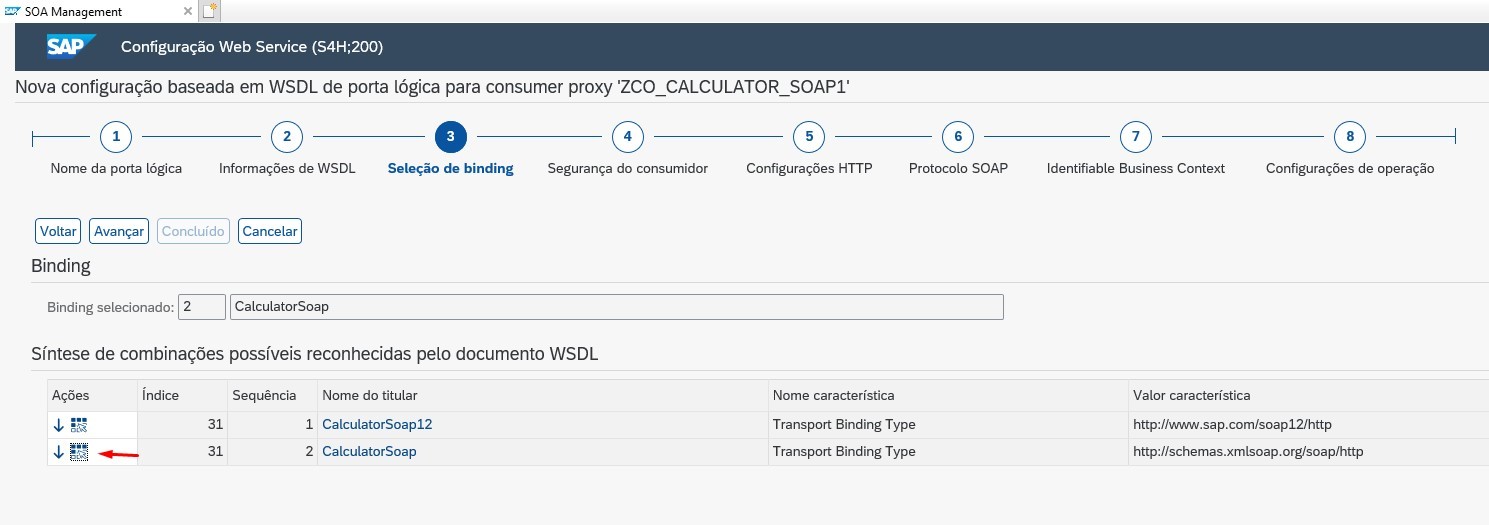
Preencha nome da porta lógica (Sugiro o **ZLP**como identificador no inicio do nome ).



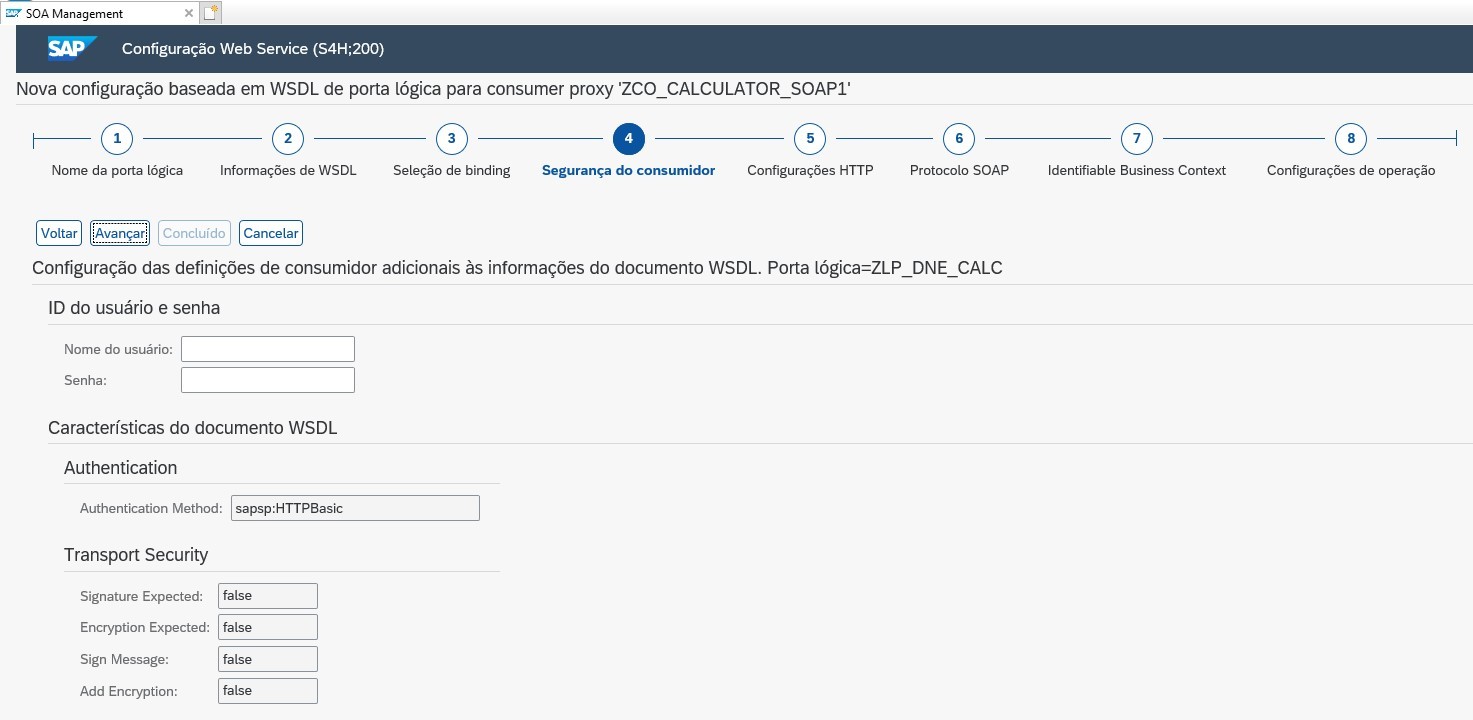
Preencha a **URL**do **WSDL**(Caso tenha usuário e senha é possível preencher também) e clique em '**Avançar**'.



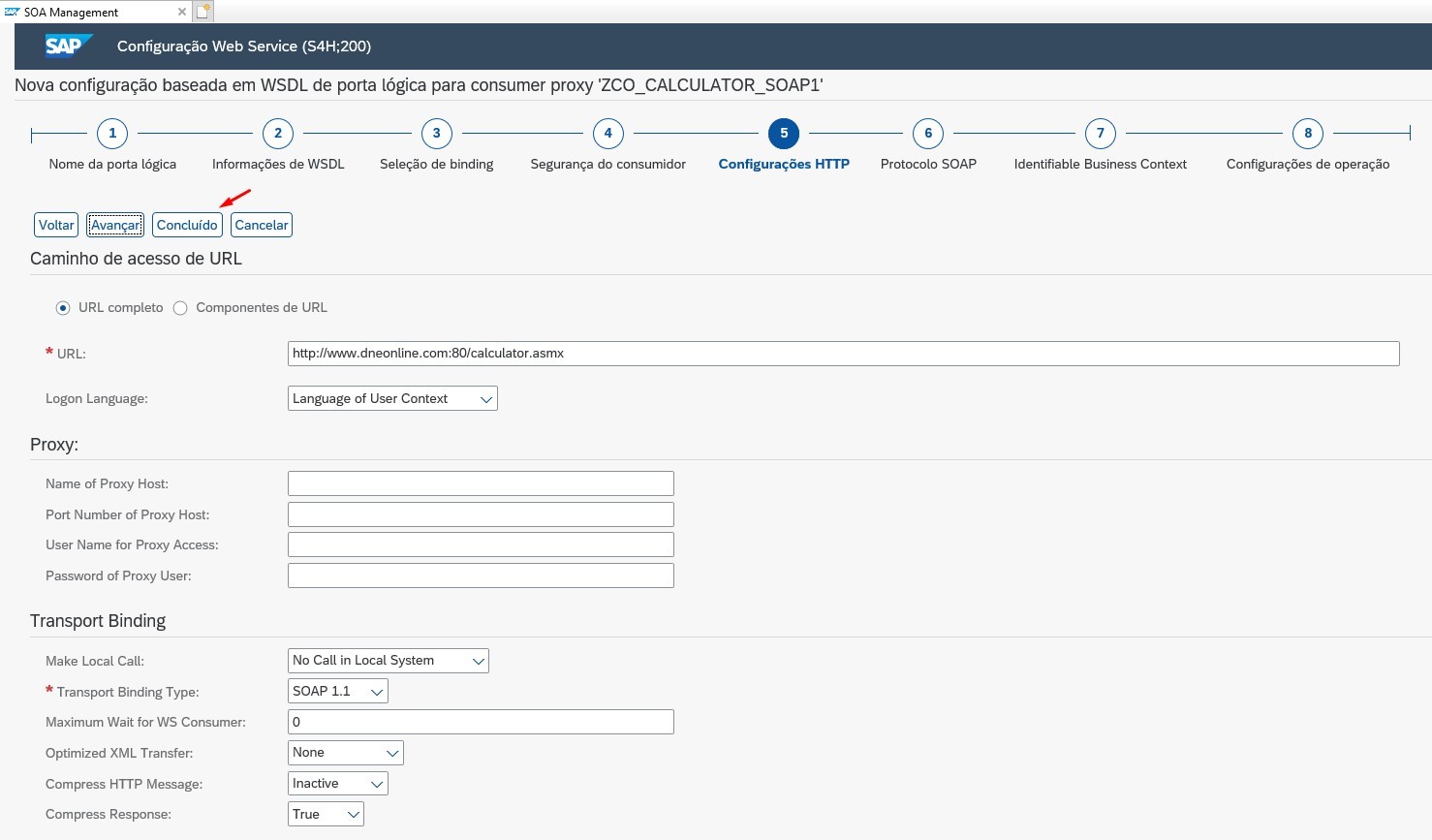
Caso tenha mais de uma definição de **Binding**selecione a que deseja usar (No nosso caso será a **CalculatorSoap**) e clique em Avançar.



Caso seja necessário alterar parâmetros de **autenticação**, pode fazer nessa tela, no nosso exemplo só vamos clicar em '**Avançar**'.



Para nosso exemplo não precisamos preencher as próximas etapas, podemos já clicar em '**Concluir**'.

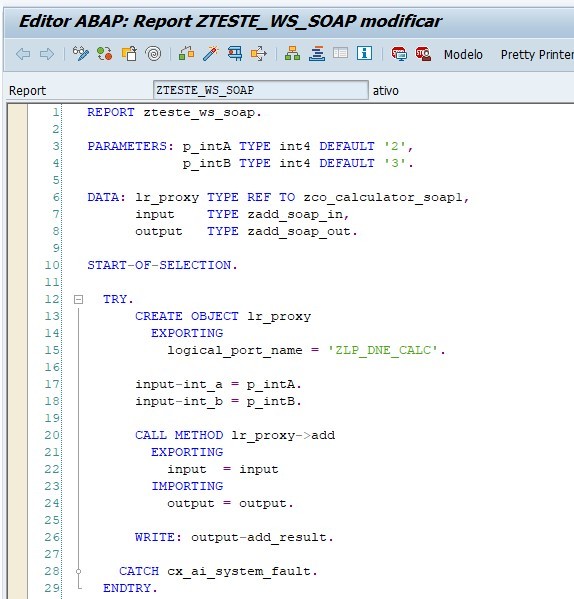


Irá aparecer essa tela e o estado será '**Ativo**'.

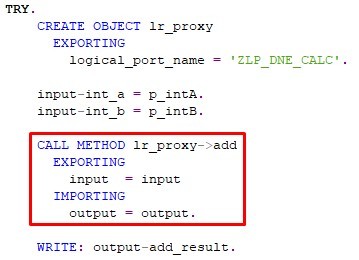


Agora voltamos ao **SAP**, como usamos isso num programa **ABAP**?

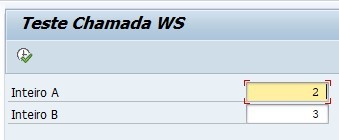
Abaixo um pequeno exemplo.



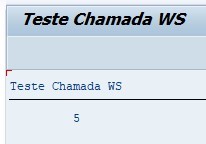
Veja que estamos usando o endpoint de soma (**add**), porém poderíamos utilizar qualquer outro, recomendo que você explore os outros métodos também.



Ao executar o programa e informar dois valores inteiros

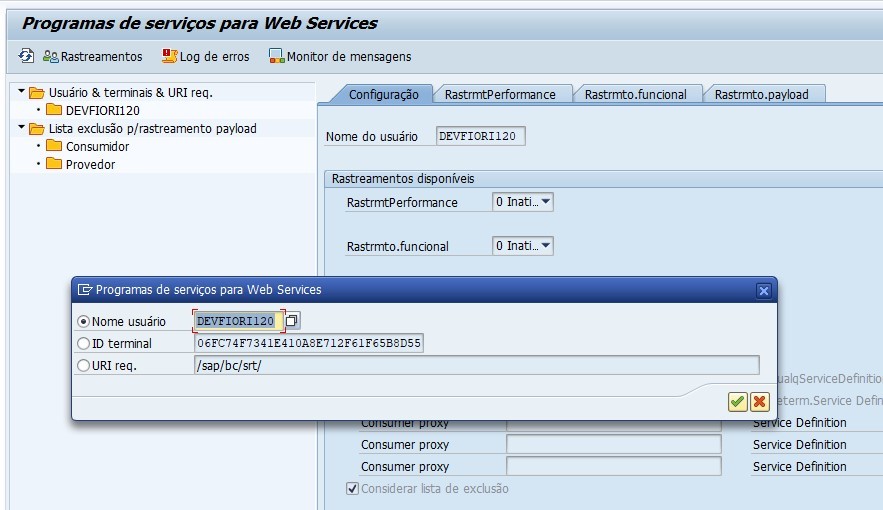


Temos o resultado em tela.

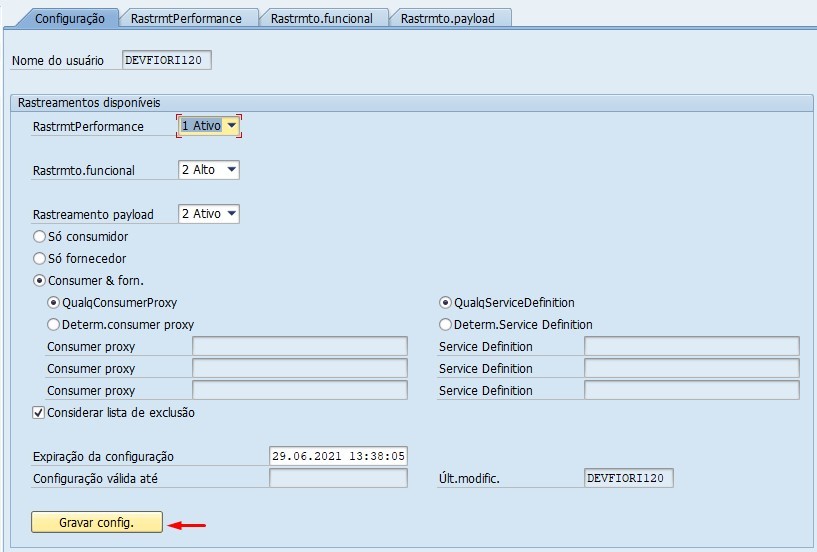


Agora como podemos ver as **mensagens**que estão **trafegando**?

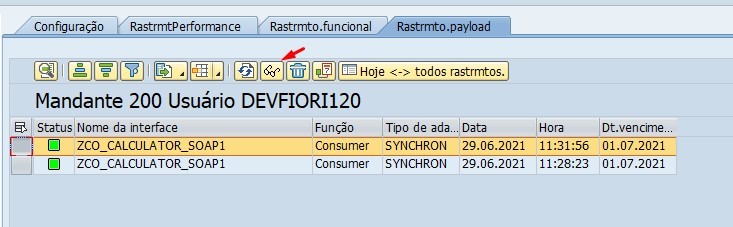
Podemos ligar um rastreamento na transação **SRT\_UTIL**clicando em '**Rastreamentos**' e selecionando se queremos ativar por usuário, terminal ou endpoint, no nosso caso estamos fazendo por usuário.



Marcamos como '**Ativo**' o que queremos que fique salvo e clicamos em '**Gravar config**.'.

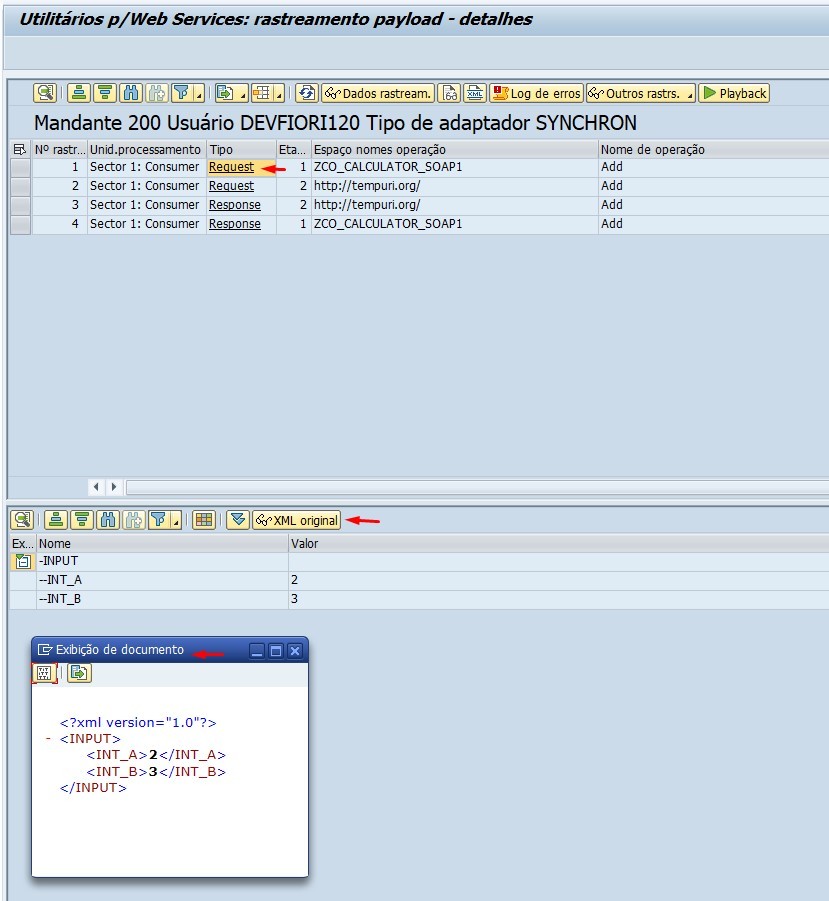


Com isso podemos ver por exemplo os **payloads** que trafegaram entre o **SAP**e o **WebService**.

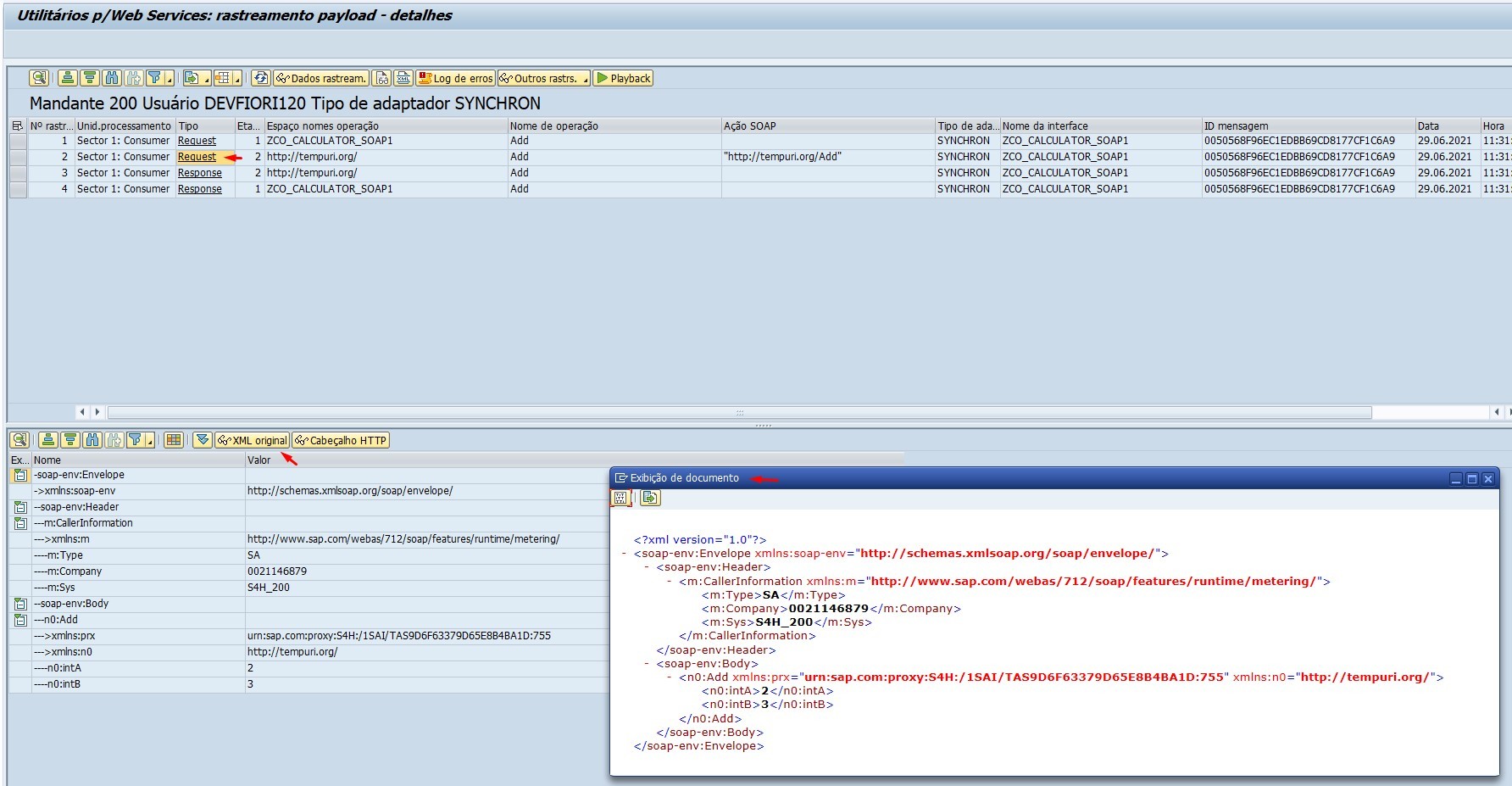


Podemos selecionar entre o que foi **enviado**e o que foi **recebido**.

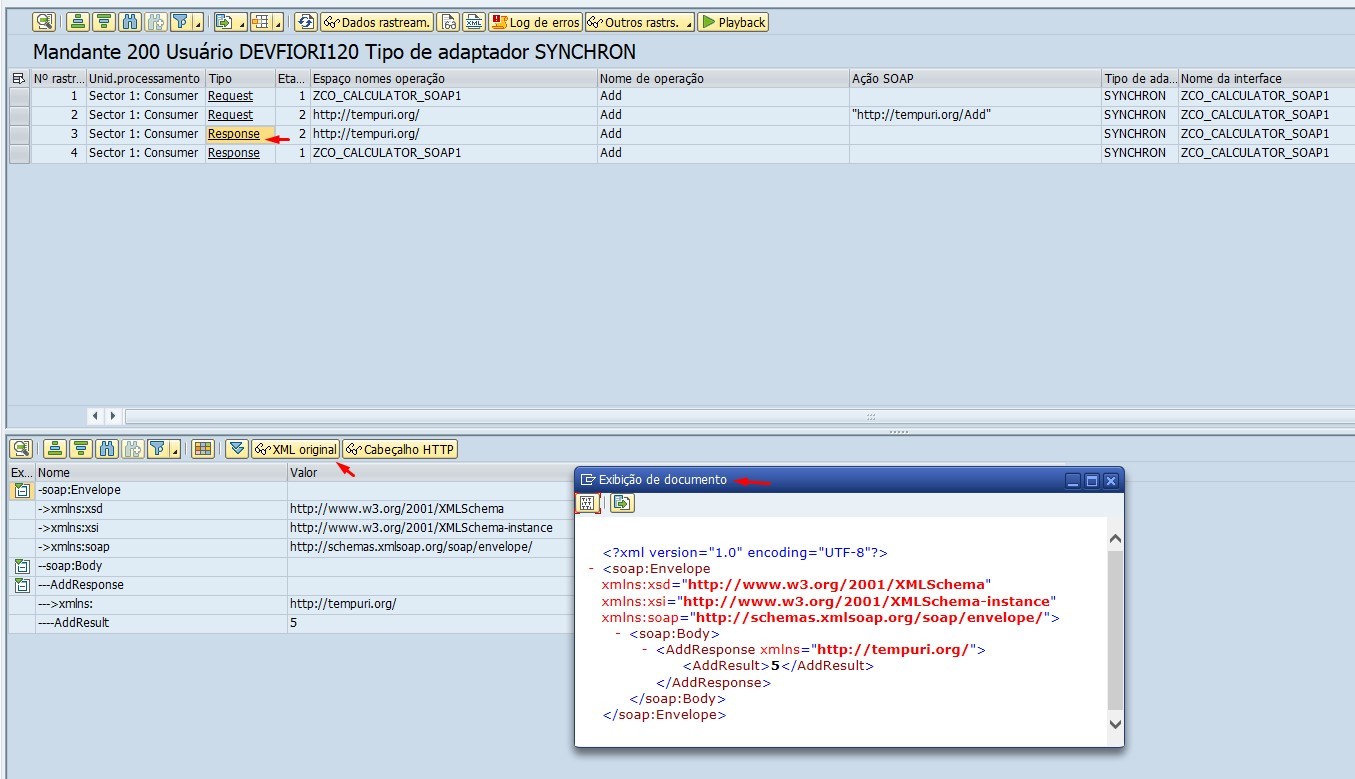
O que saiu do SAP.



O que entrou no Webservice.



O que saiu do WebService.



E o que entrou no SAP.

