FIAP

MBA+

MBA em Desenvolvimento de Aplicações Java - SOA e Internet das Coisas

FIAP

MBA+

Mule

Laboratório 5

Prof. André Pereira, MSc


Source:http://blog.comshalom.org/projecoesdefe/x-men-dias-futuro-esquecido/
"Só porque alguém tropeça e erra o caminho, não quer dizer que está perdido para sempre"

Agenda

Agenda

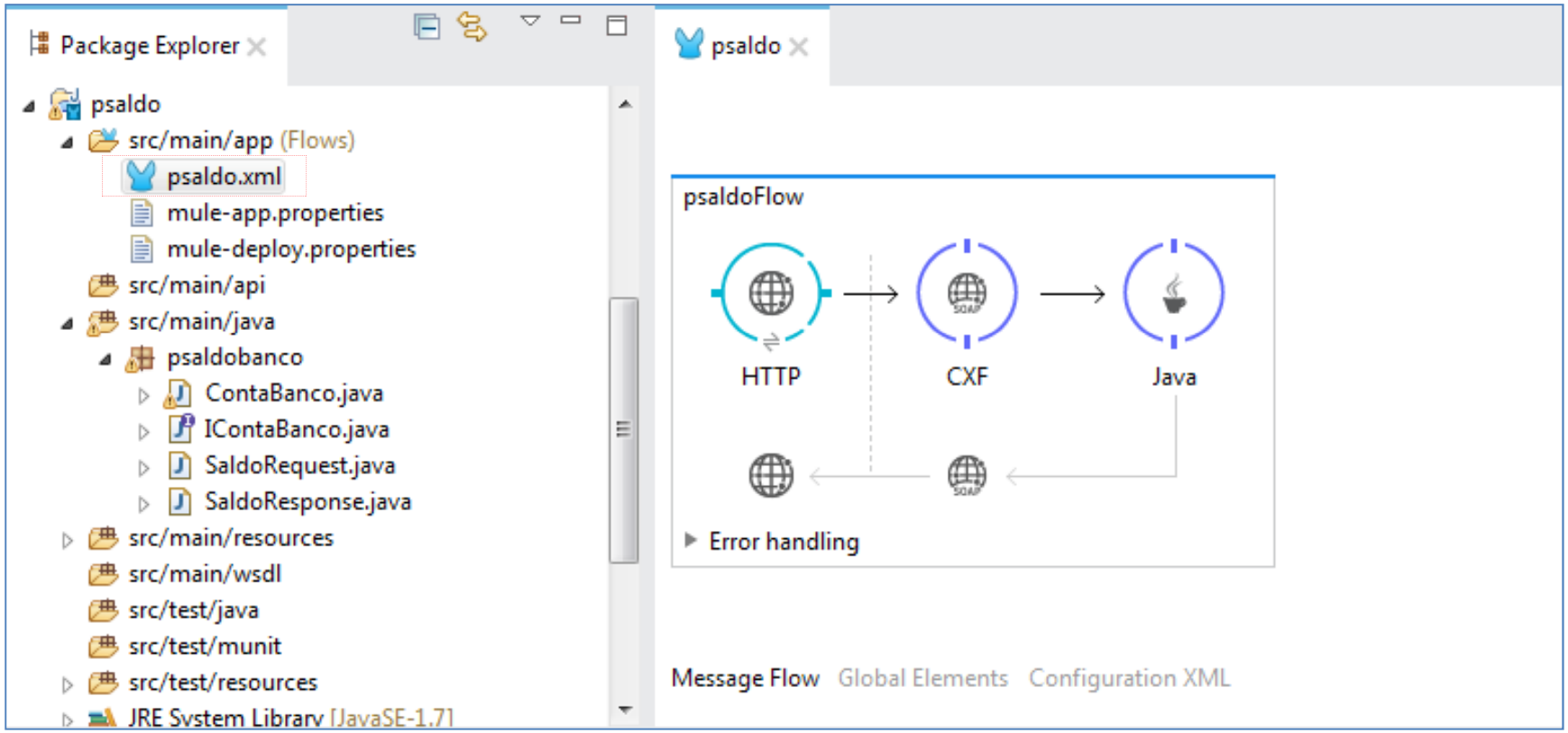
FIAP

- O Que será feito nesse Lab.
  - Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”.
  - Testando o serviço “Proxy” no Mule Runtime e SoapUI.

Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

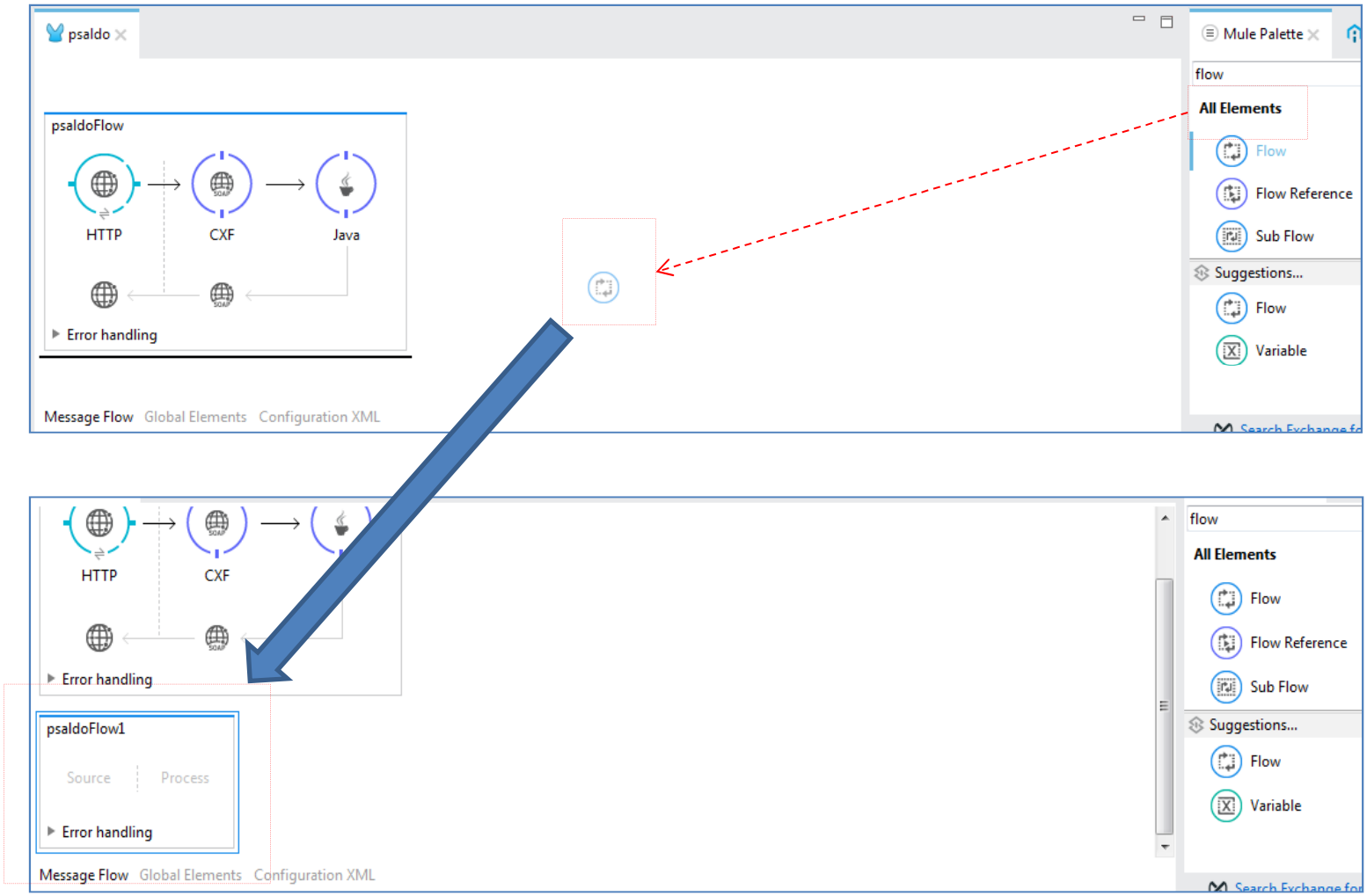
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

1) Vamos aumentar o projeto “Saldo” criando um serviço de “Proxy” contendo dados alterados. Se o mesmo não estiver aberto, abra o projeto “psaldo.xml”.



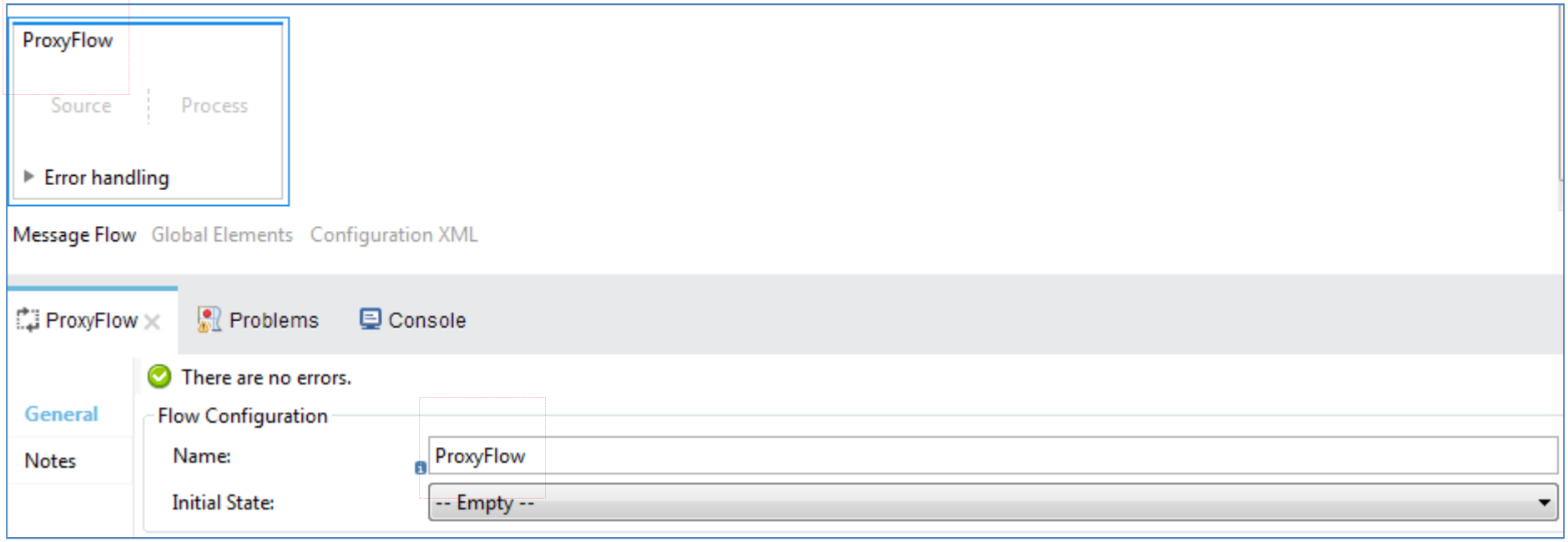
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

2) Vá até a paleta e escolha o componente “Flow”. Clique e arraste até a área de trabalho conforme a figura abaixo.



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

3) Troque o nome do novo fluxo para “ProxyFlow” conforme a figura abaixo.

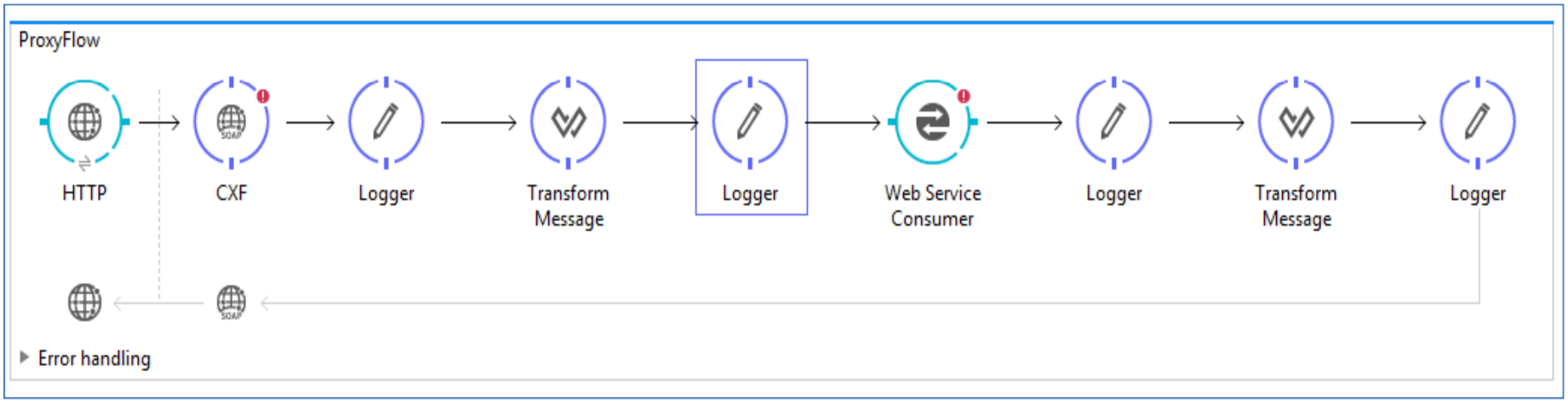


Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

4) Vamos criar a lógica do proxy. Arraste as seguintes quantidades de componentes:

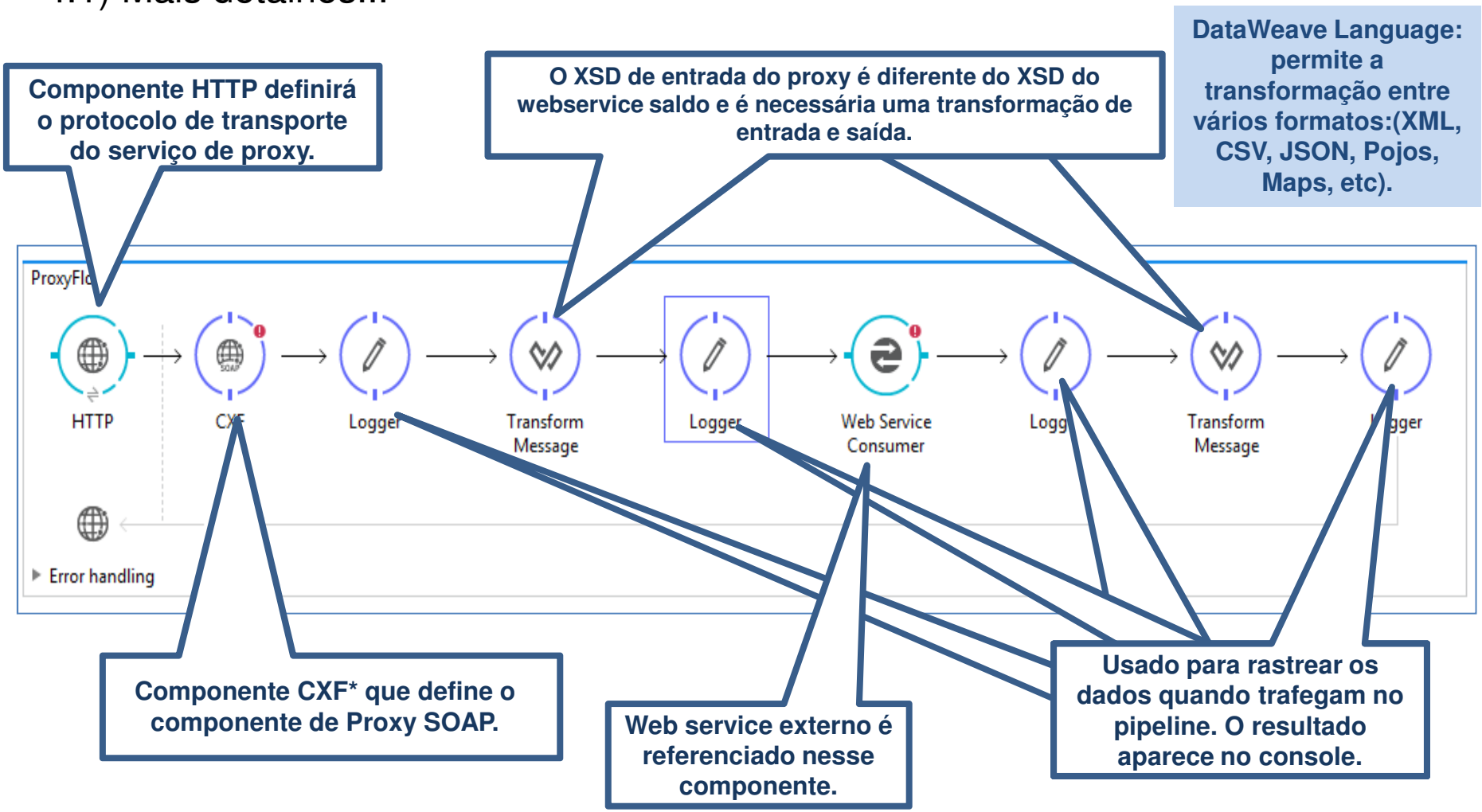
- 1 componente “HTTP”
- 4 componentes “Logger”
- 2 componentes “Transform Message”
- 1 componente “Web Service Consumer”

para o “ProxyFlow” de acordo com a sequencia abaixo:



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

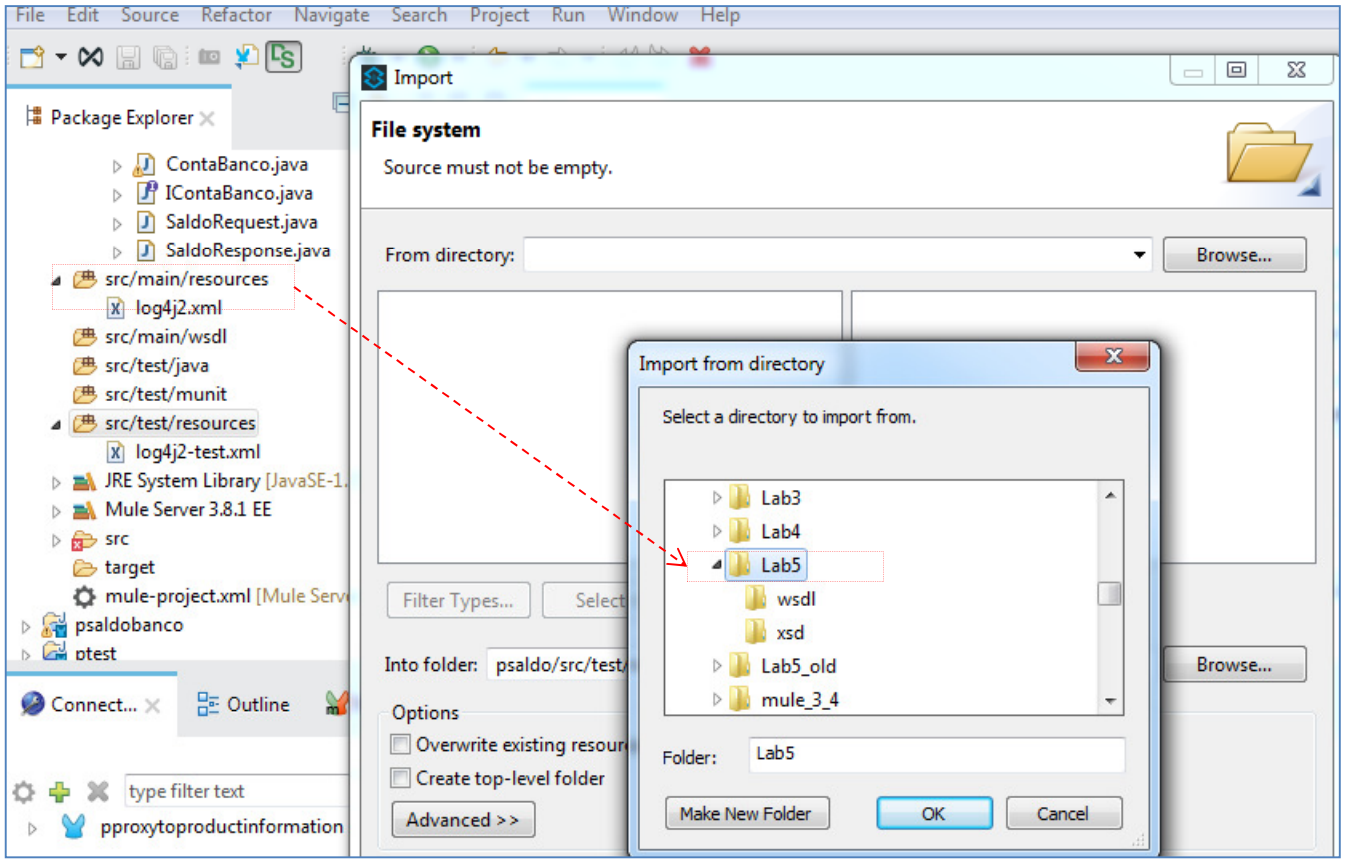
4.1) Mais detalhes...



Apache CXF* (projetos "Celtix" and "XFire".) é um framework open source que dá suporte a criação e consumo de Web Services e está de acordo com JAX-WS e JAX-RS. Suporta: SOAP, XML/HTTP, RESTful HTTP ou CORBA. Um Componente Proxy permite implementar padrões WS* tais como: WS-Security ou WS-Addressing.

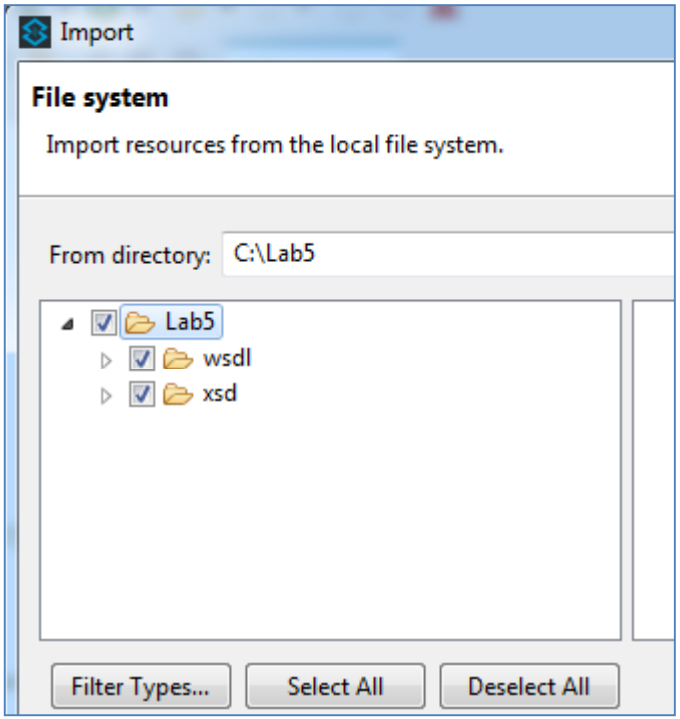
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

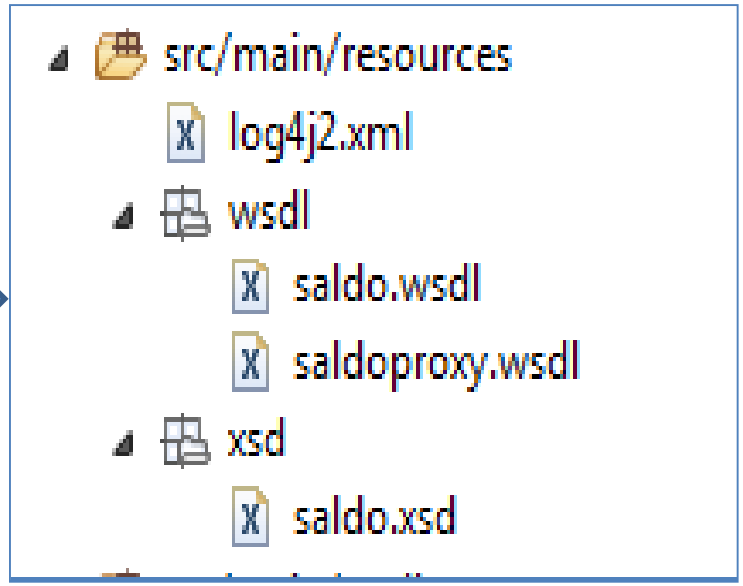
5) O novo serviço de proxy, utilizará como base o mesmo wsdl gerado pelo serviço “saldo”, contudo, ele também utilizará um novo arquivo de xsd para utilizar como metadado. Como esse metadado é diferente do metadado original pertencente ao serviço “saldo”, necessitará das transformações. Primeiramente importe o wsdl e xsd que se encontra na pasta Lab5 conforme a figura abaixo:



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

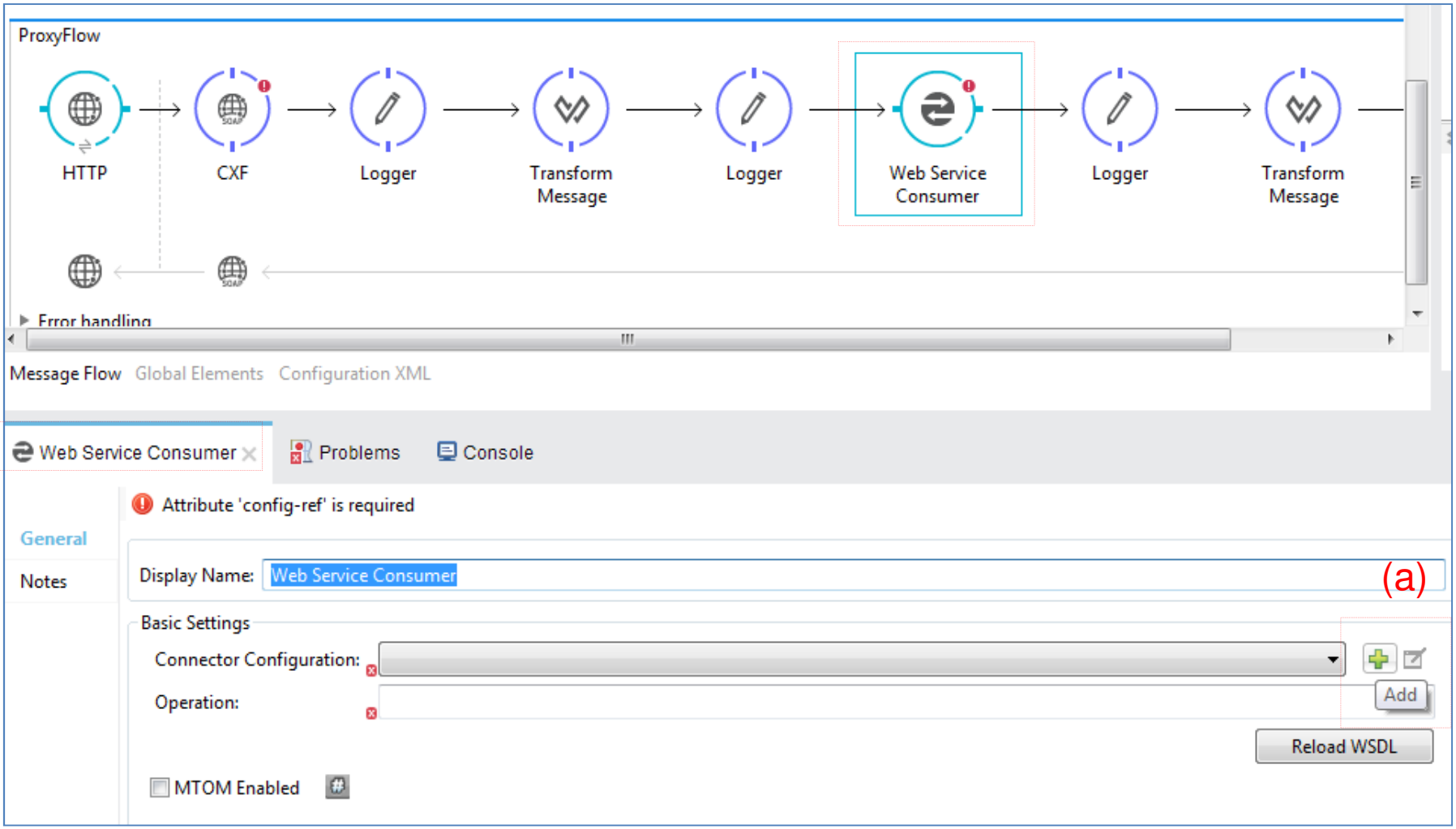
6) Os arquivos serão importados para o pacote: “src/main/resources”.





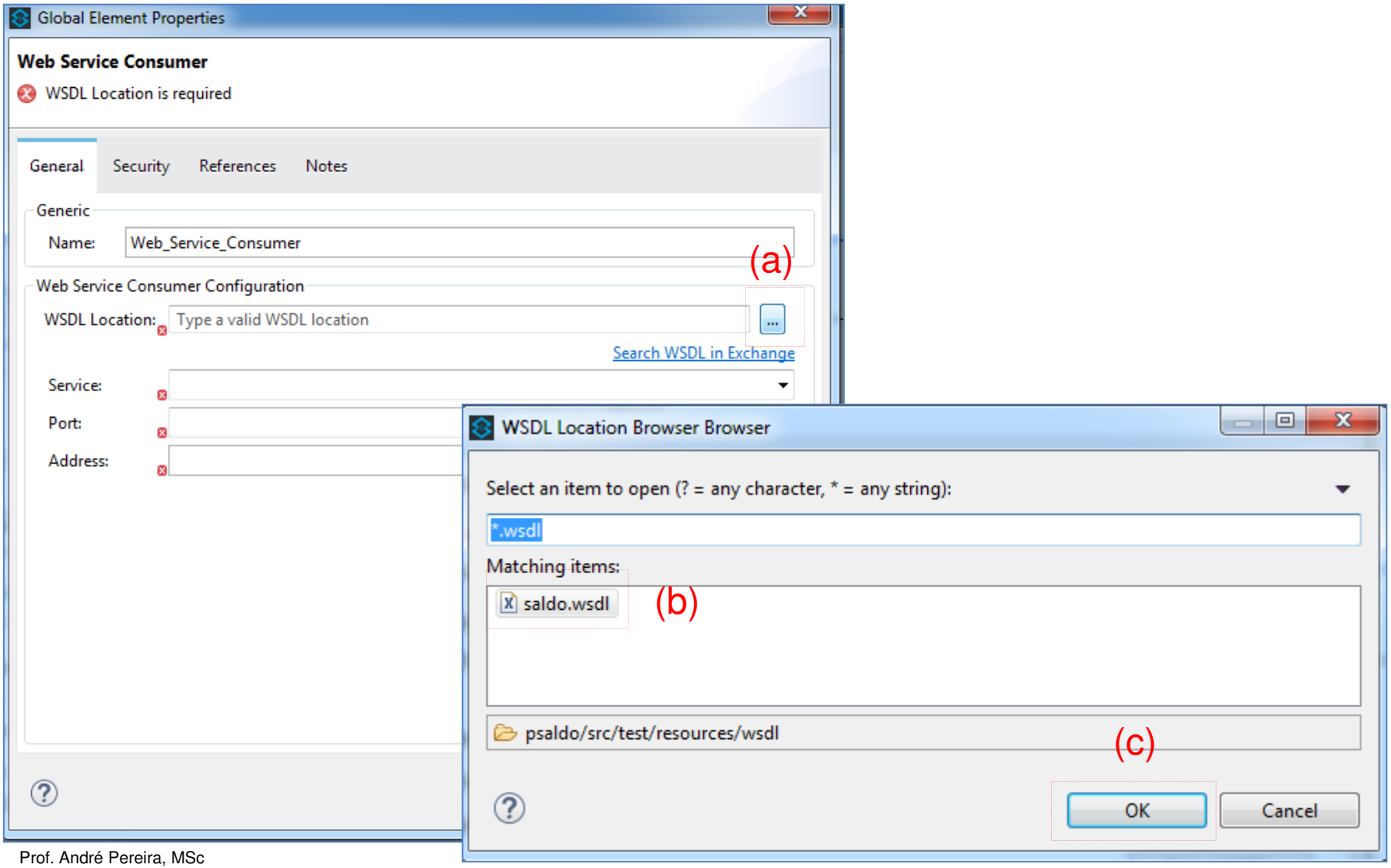
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

7) Agora começaremos a configurar cada componente. Clique em “Web Service Consumer” e no clique no sinal de “+ verde” (a).



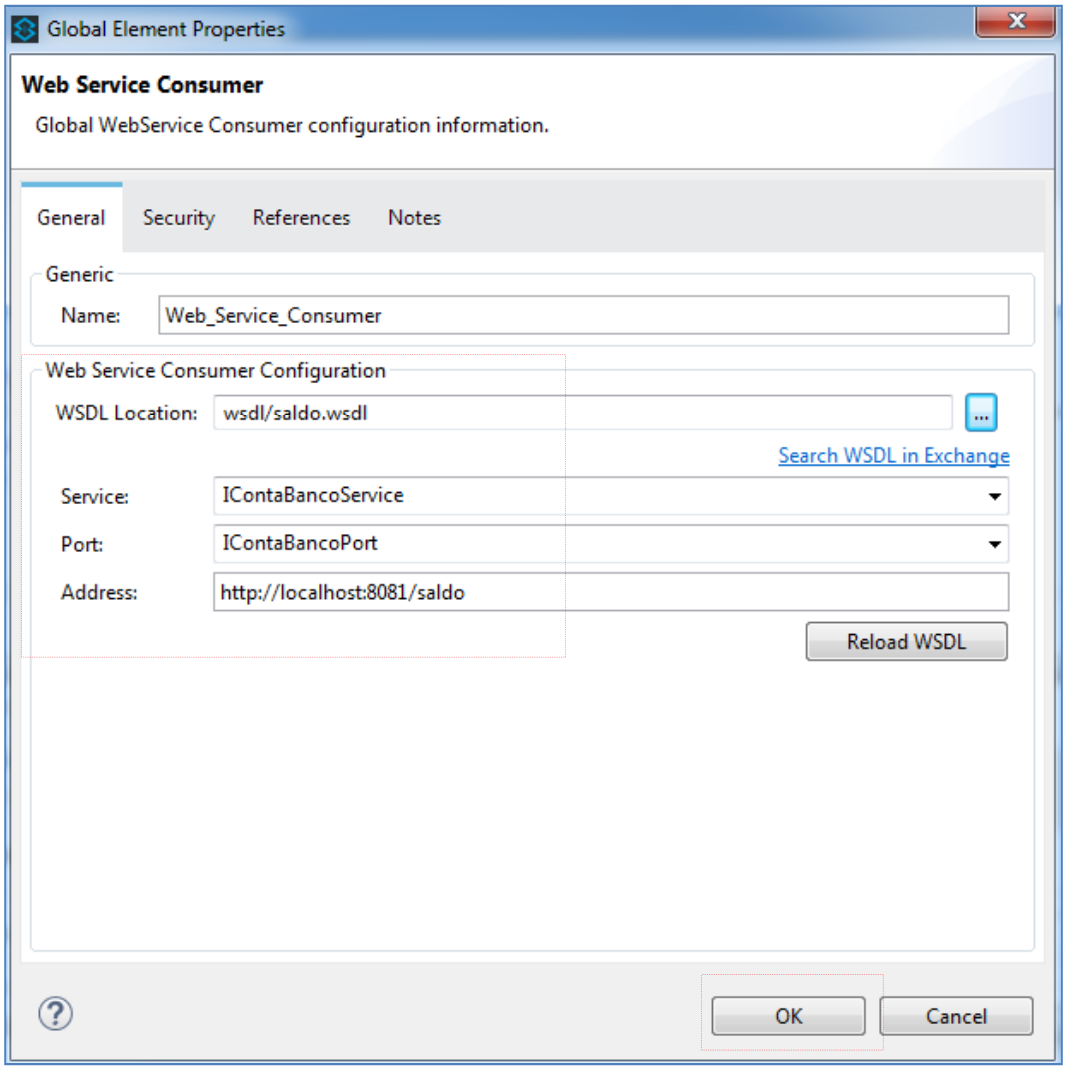
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

8) Na tela “Global Element Properties”, clique em (a) para inserir o wsdl importado. Selecione “saldo.wsdl” (b) e depois clique em “OK”. (c)



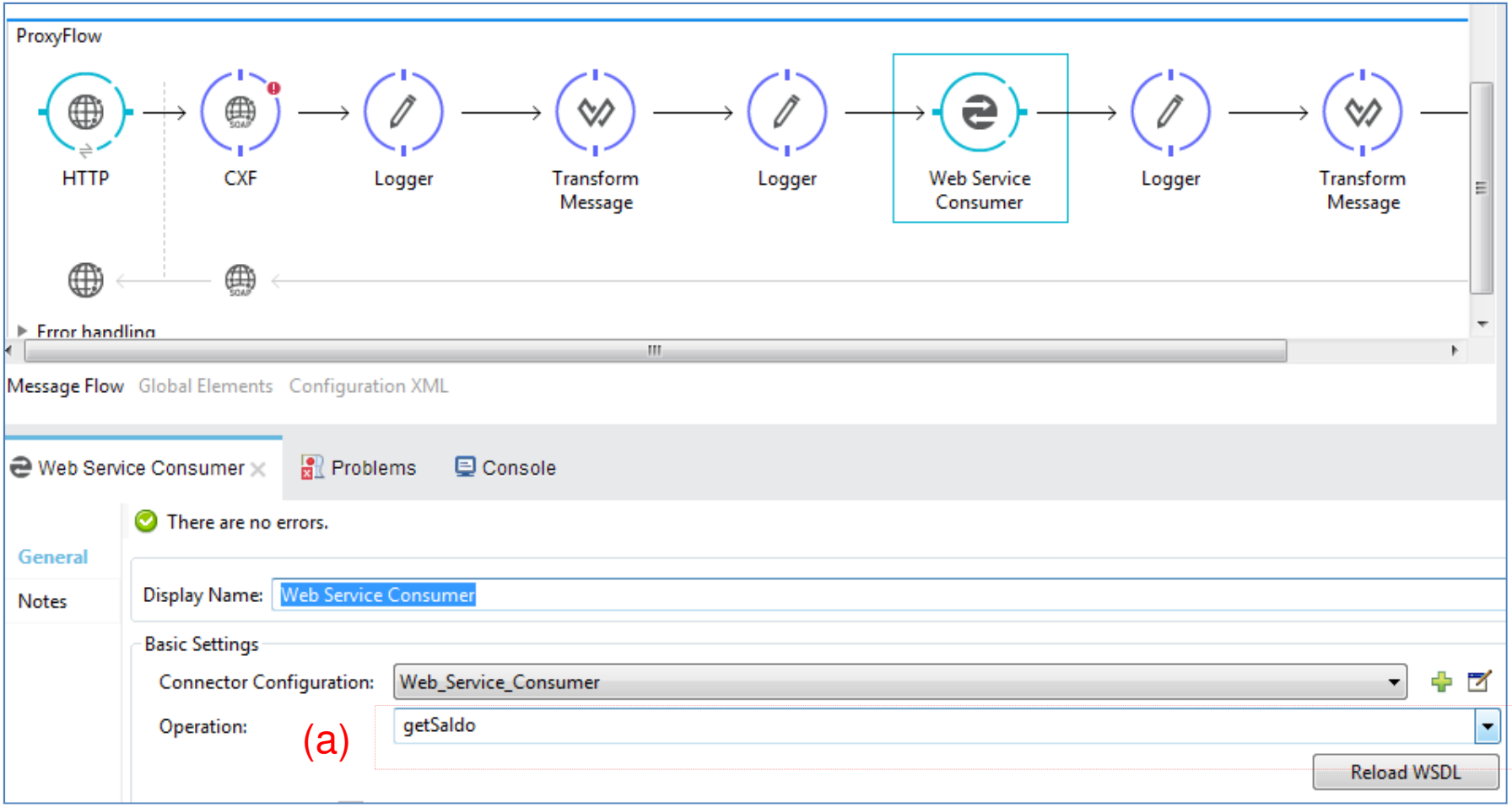
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

9) Os demais campos serão automaticamente preenchidos. Aguarde um pouco e depois clique em “OK”.



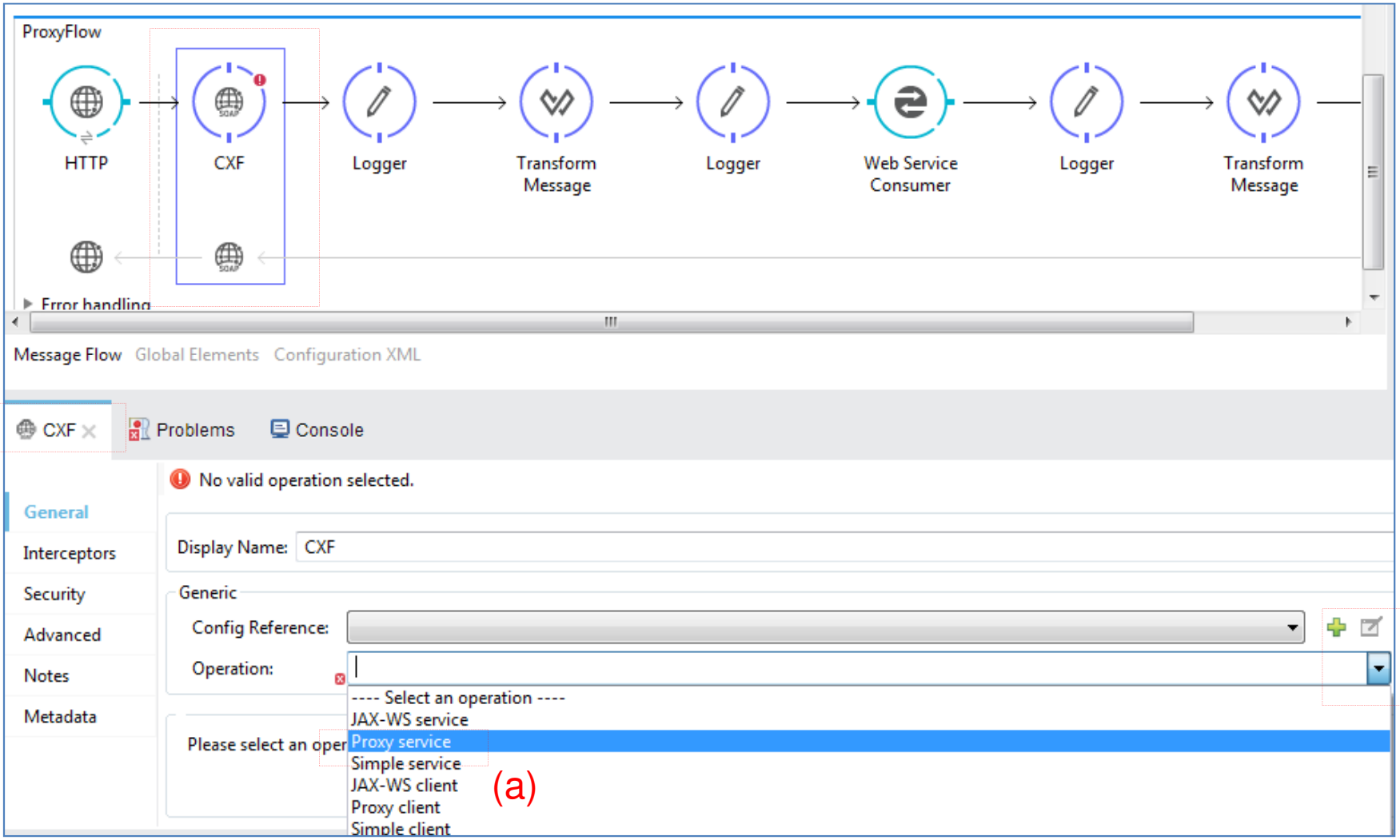
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

10) Selecione a operação “getSaldo” (a) e depois salve.



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

11) Selecione o componente “CXF” depois selecione “Proxy service” (a).

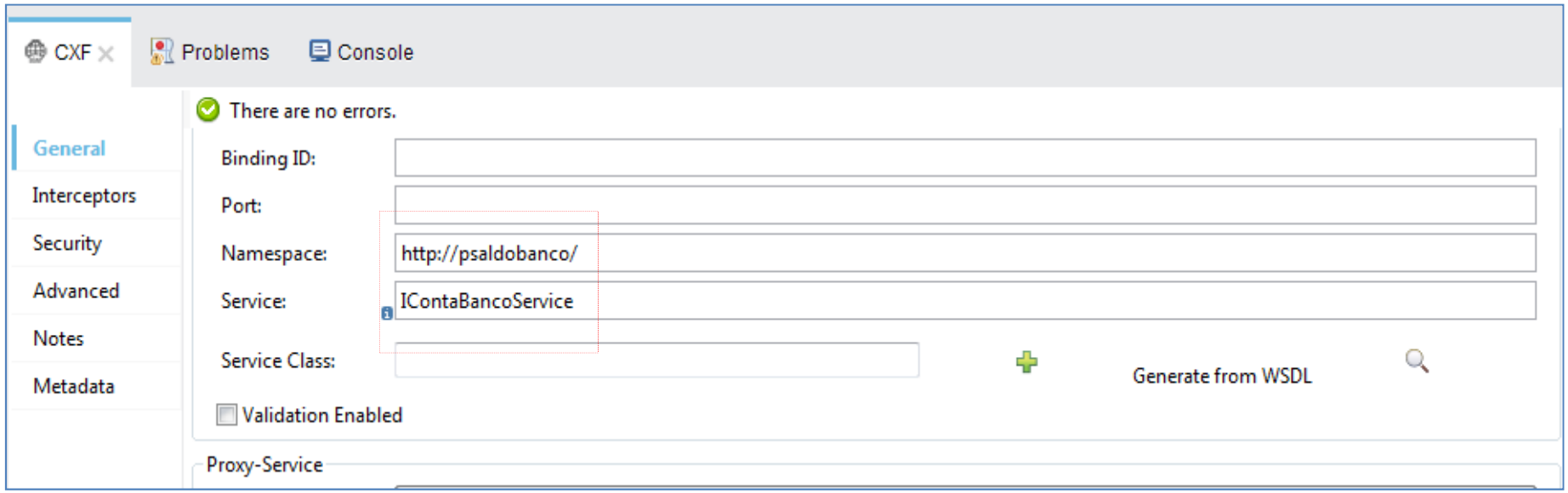


Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

12) Abaixo, preencha as informações:

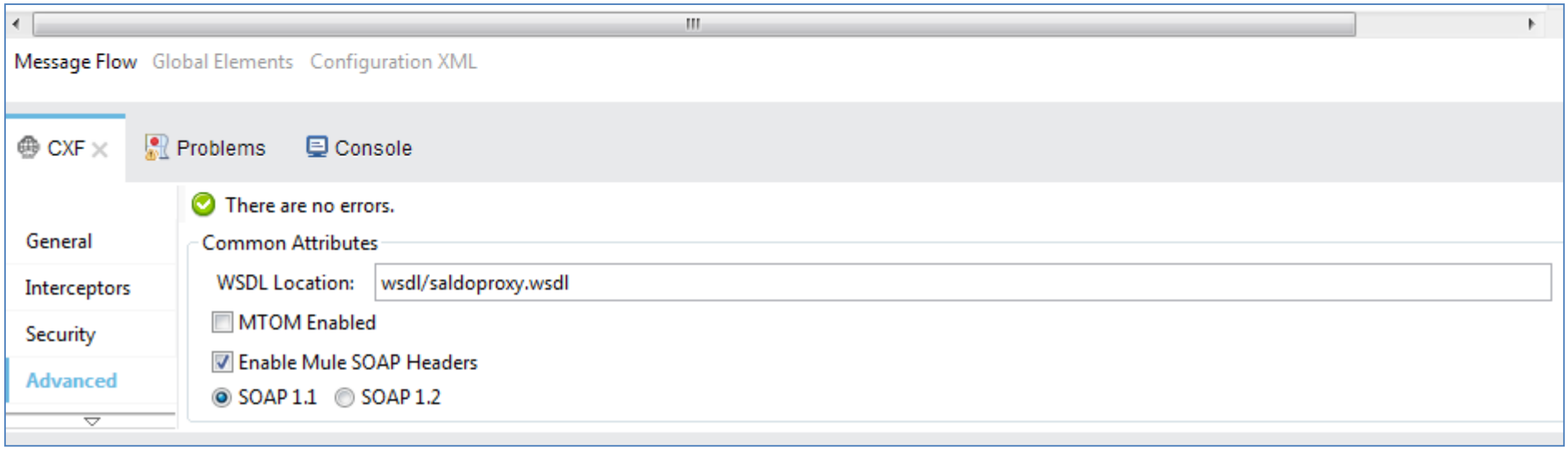
- Namespace: http://psaldobanco/
- Service: IContaBancoService

De acordo com a figura abaixo.



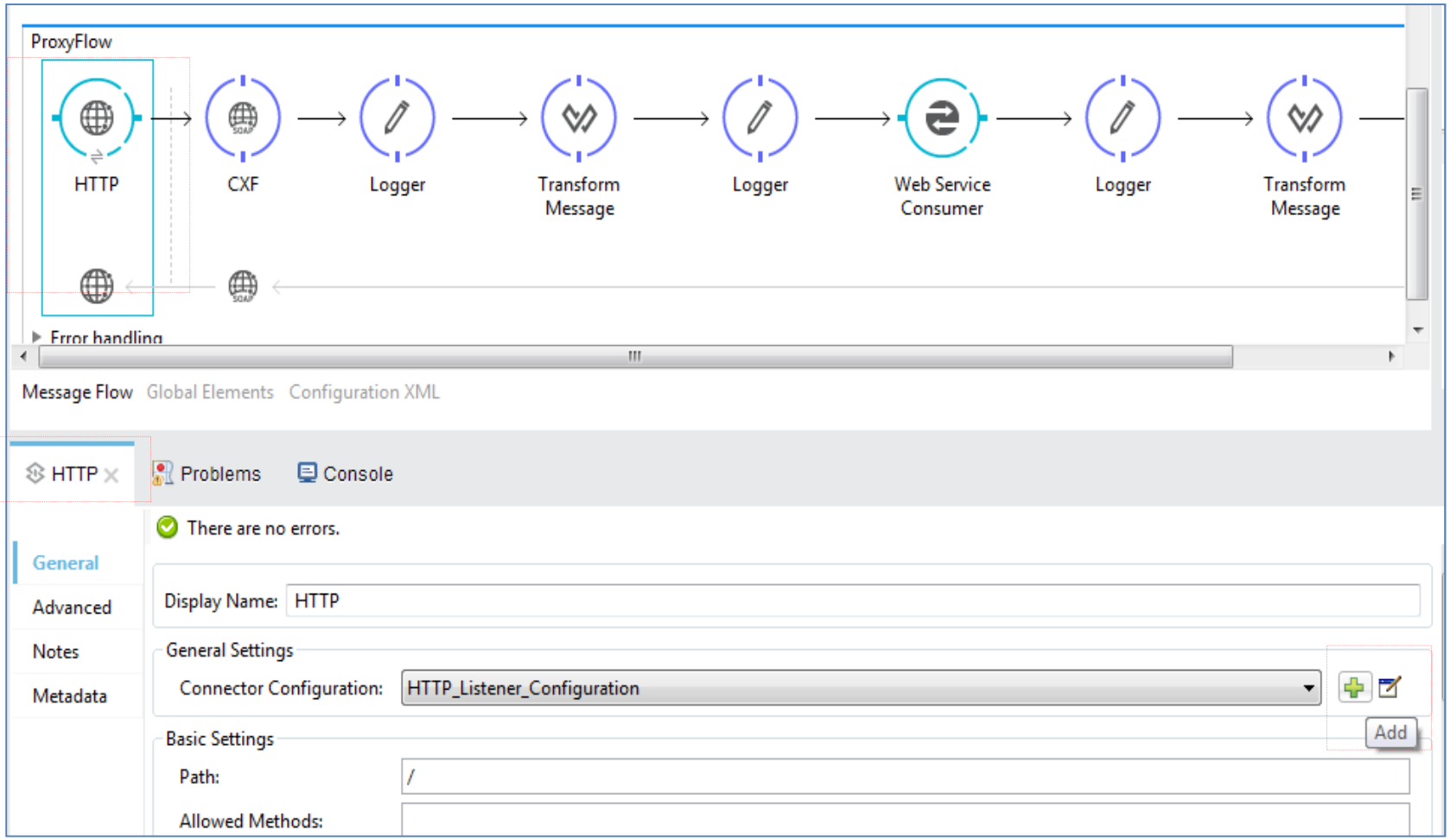
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

13) Depois vá até a aba “Advanced” e insiria no campo “WSDL Location” o valor: “wsdl/saldoproxy.wsdl” conforme figura abaixo. Não esqueça de salvar.



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

14) Selecione o componente HTTP. Depois adicione um novo “Connector Configuration” clicando no sinal de “+ verde” (a).



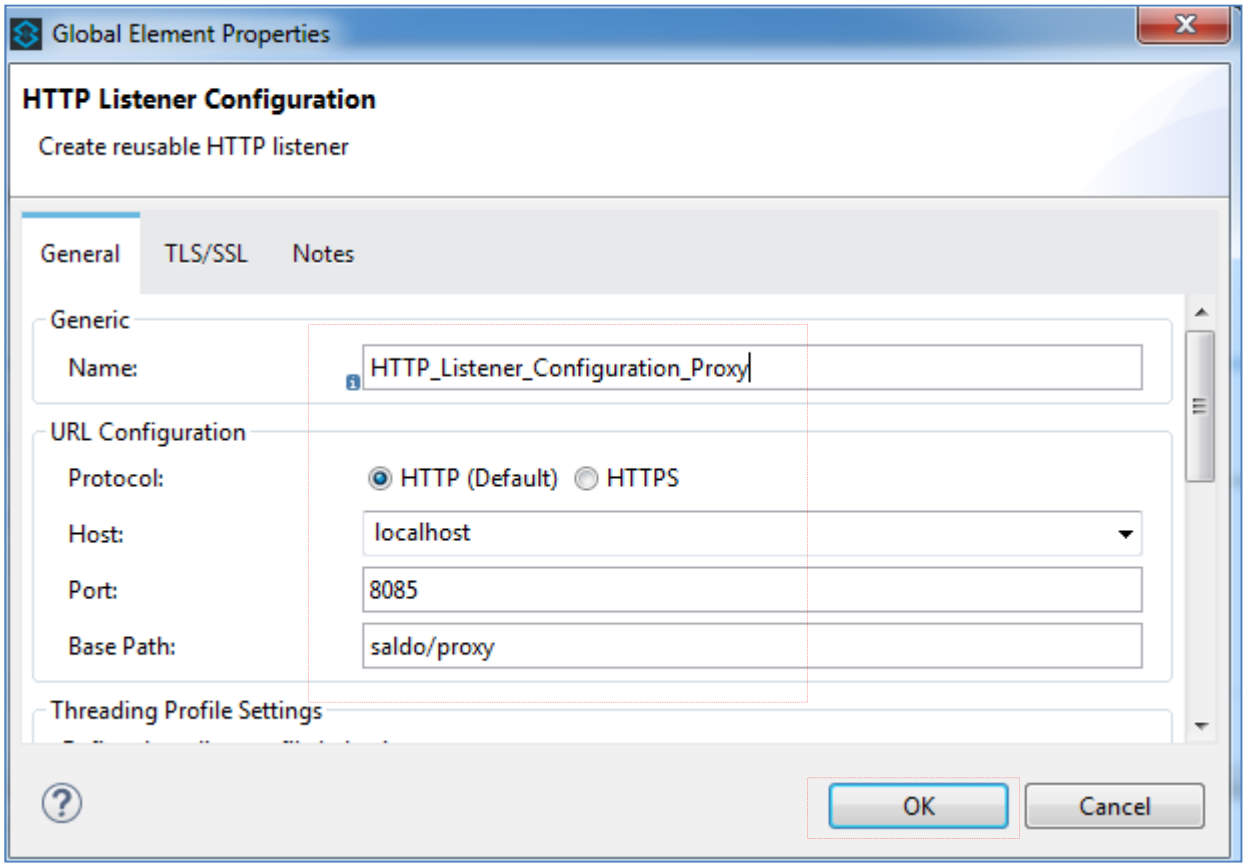
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

15) Em “Global Element Properties”, preencha os seguintes valores:

- Name: HTTP_Listener_Configuration_Proxy
- Host: localhost
- Port: 8085
- Base Path: saldo/proxy

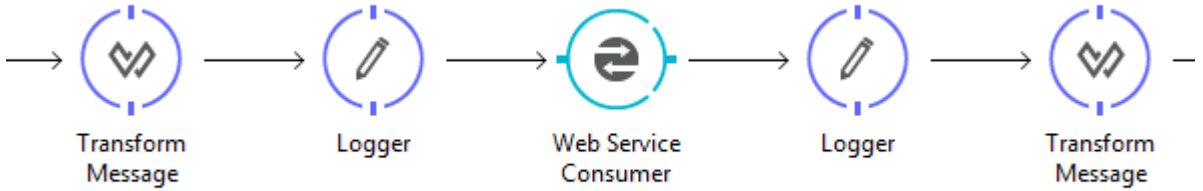
De acordo com afigura abaixo.

NÃO ESQUEÇA DE SALVAR!!!



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

16) O Próximo passo é a configuração dos componentes de transformação. Observem abaixo que antes e depois da invocação do Webservice “saldo”, acontece uma transformação. Repare que na ida os tipos são diferentes e na volta, existem campos à mais.



Primeira transformação

```
<xs:complexType name="saldoRequest">
  <xs:sequence>
    <xs:element minOccurs="0" name="agencia" type="xs:string"/>
    <xs:element minOccurs="0" name="conta" type="xs:string"/>
    <xs:element minOccurs="0" name="name" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="getSaldoResponse">
  <xs:sequence>
    <xs:element minOccurs="0" name="return" type="tns:saldoResponse"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="saldoRequest">
  <xs:sequence>
    <xs:element minOccurs="0" name="agencia" type="xs:integer"/>
    <xs:element minOccurs="0" name="conta" type="xs:integer"/>
    <xs:element minOccurs="0" name="name" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="getSaldoResponse">
  <xs:sequence>
    <xs:element minOccurs="0" name="return" type="tns:saldoResponse"/>
```

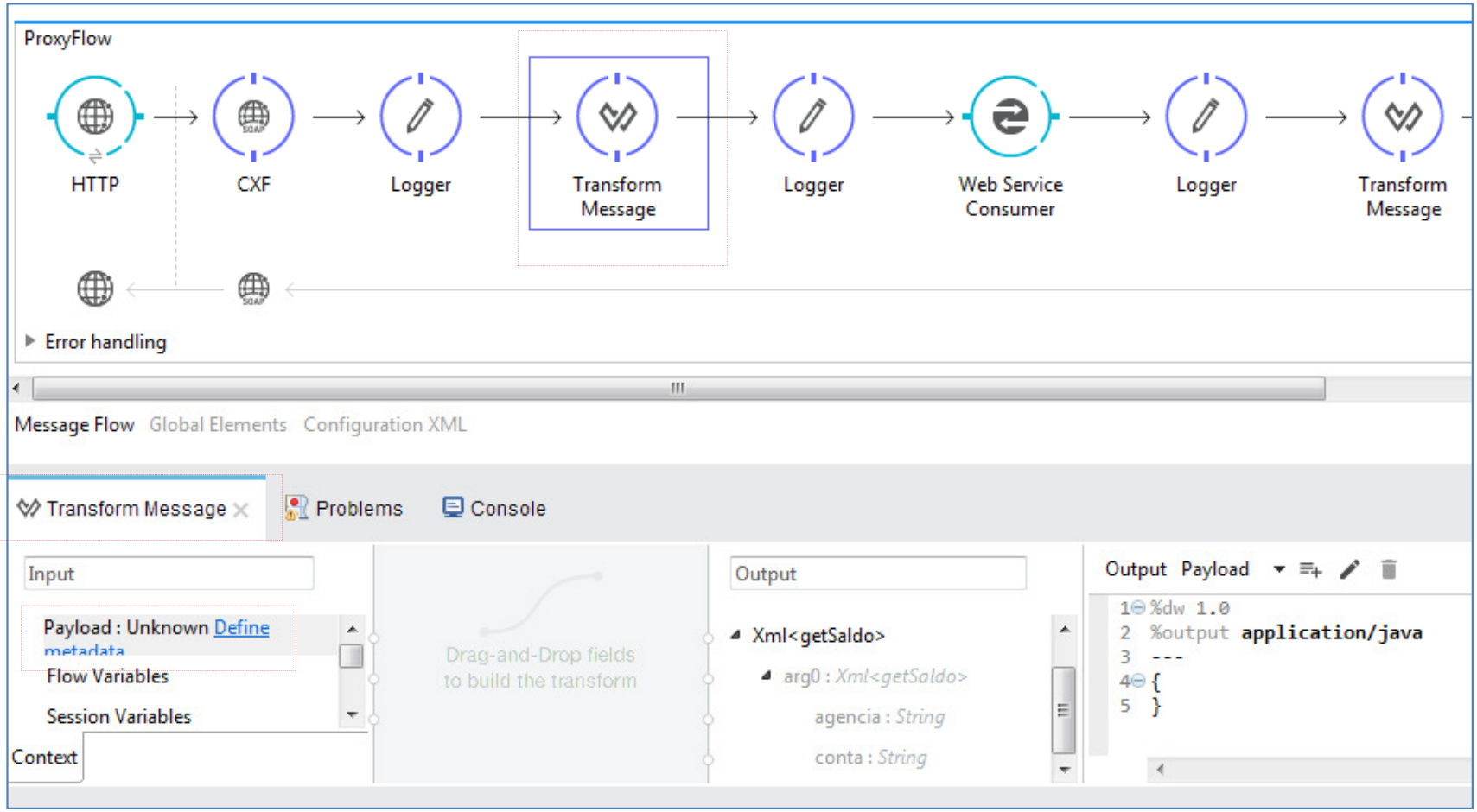
Segunda transformação

```
<xs:complexType name="saldoResponse">
  <xs:sequence>
    <xs:element minOccurs="0" name="agencia" type="xs:string"/>
    <xs:element minOccurs="0" name="conta" type="xs:string"/>
    <xs:element minOccurs="0" name="date" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="name" type="xs:string"/>
    <xs:element name="saldo" type="xs:double"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:complexType name="saldoResponse">
  <xs:sequence>
    <xs:element minOccurs="0" name="agenciaConta" type="xs:string"/>
    <xs:element minOccurs="0" name="nomeAgencia" type="xs:string"/>
    <xs:element minOccurs="0" name="date" type="xs:dateTime"/>
    <xs:element minOccurs="0" name="name" type="xs:string"/>
    <xs:element name="saldo" type="xs:double"/>
    <xs:element name="currency" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

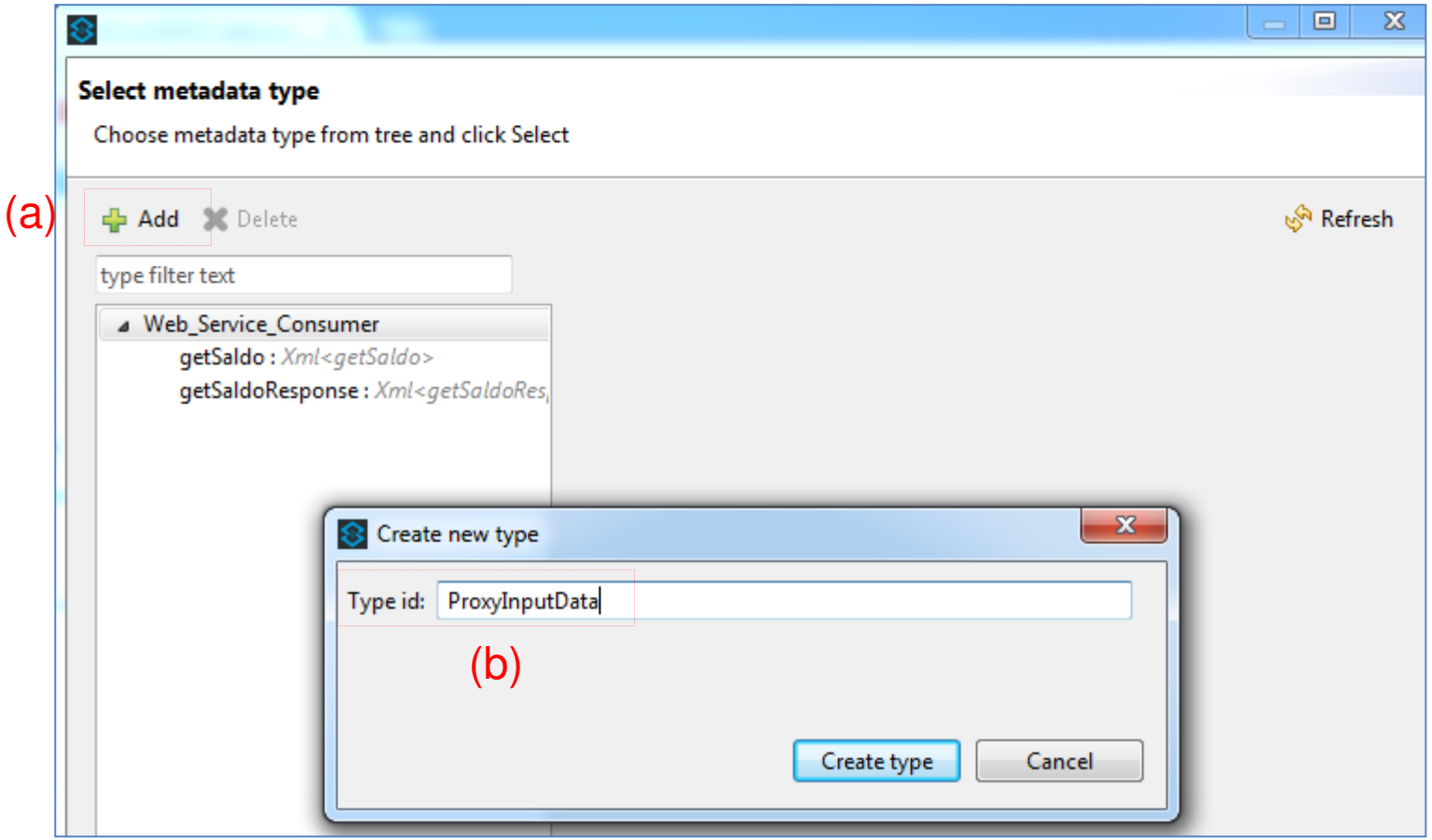
17) Clique no primeiro componente “Transform Message”. Vamos definir o metadado de entrada (que vem do proxy para a chamada do webservice). Clique em (a).



(a)

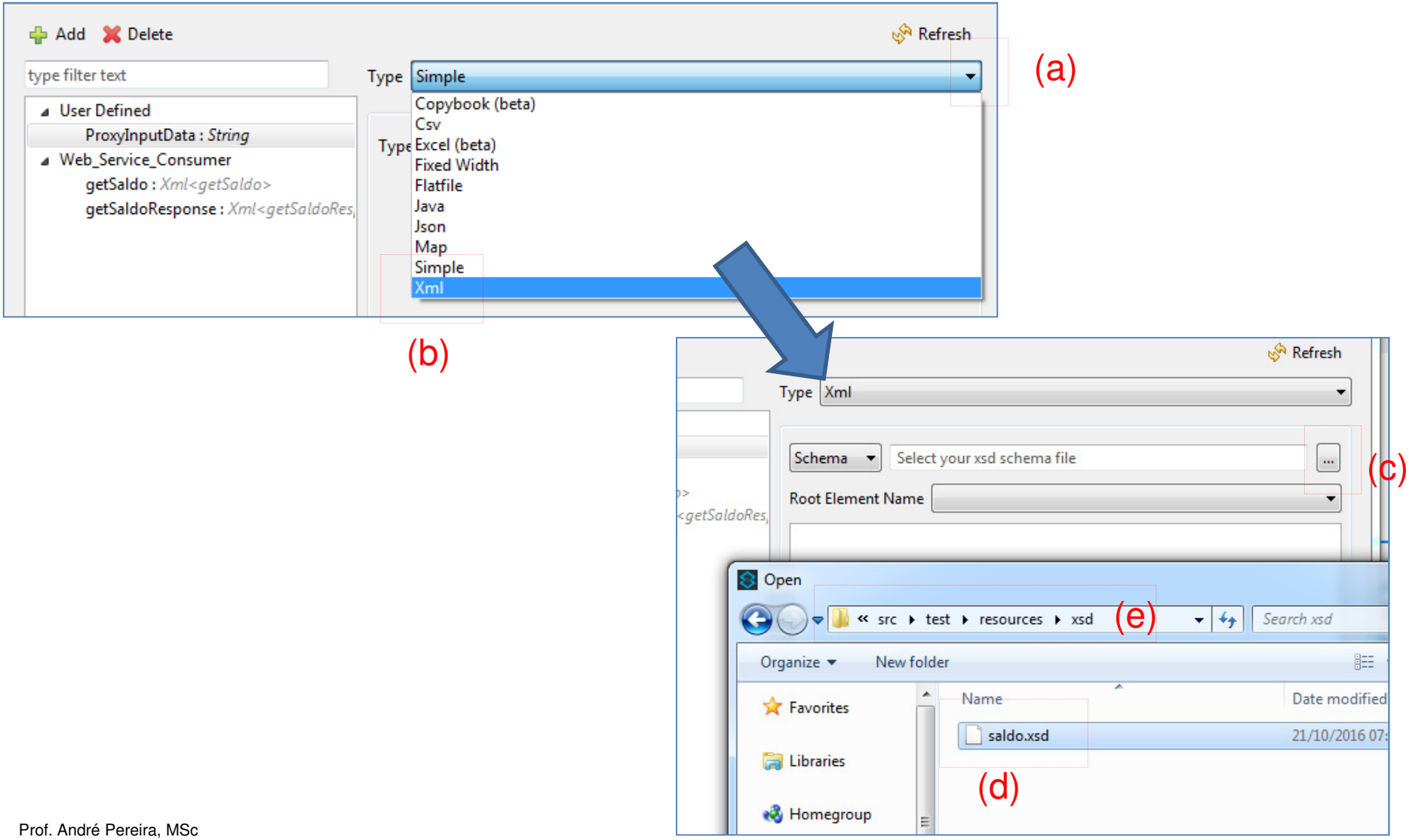
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

18) Clique no “+ verde” (a) para adicionar um novo metadado. Depois coloque o nome de “ProxyInputData” (b).



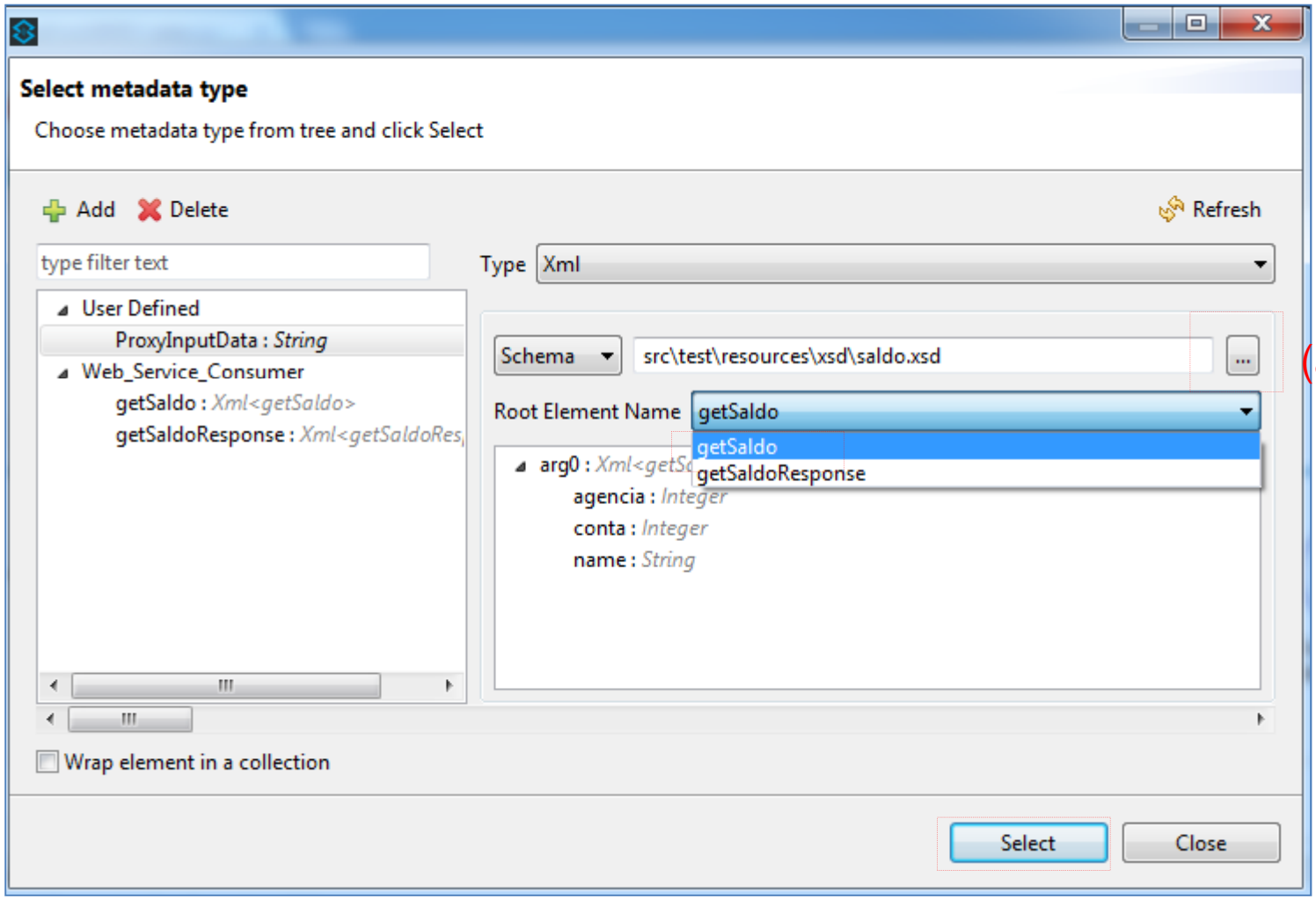
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

19) Clique em (a) e escolha “Xml” (b). Depois em (c) e escolha o arquivo “saldo.xsd” (d) que está na pasta “src/test/resources/xsd” (e).



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

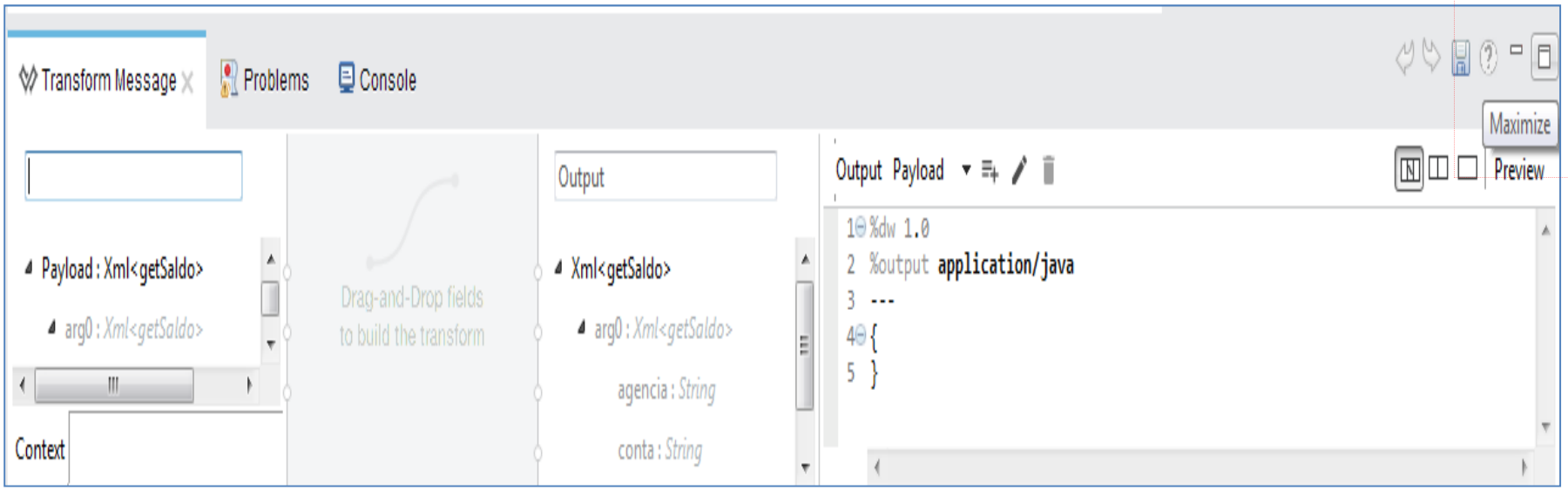
20) Clique em (a) e depois selecione “getSaldo”. Depois clique no botão “Select”.



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

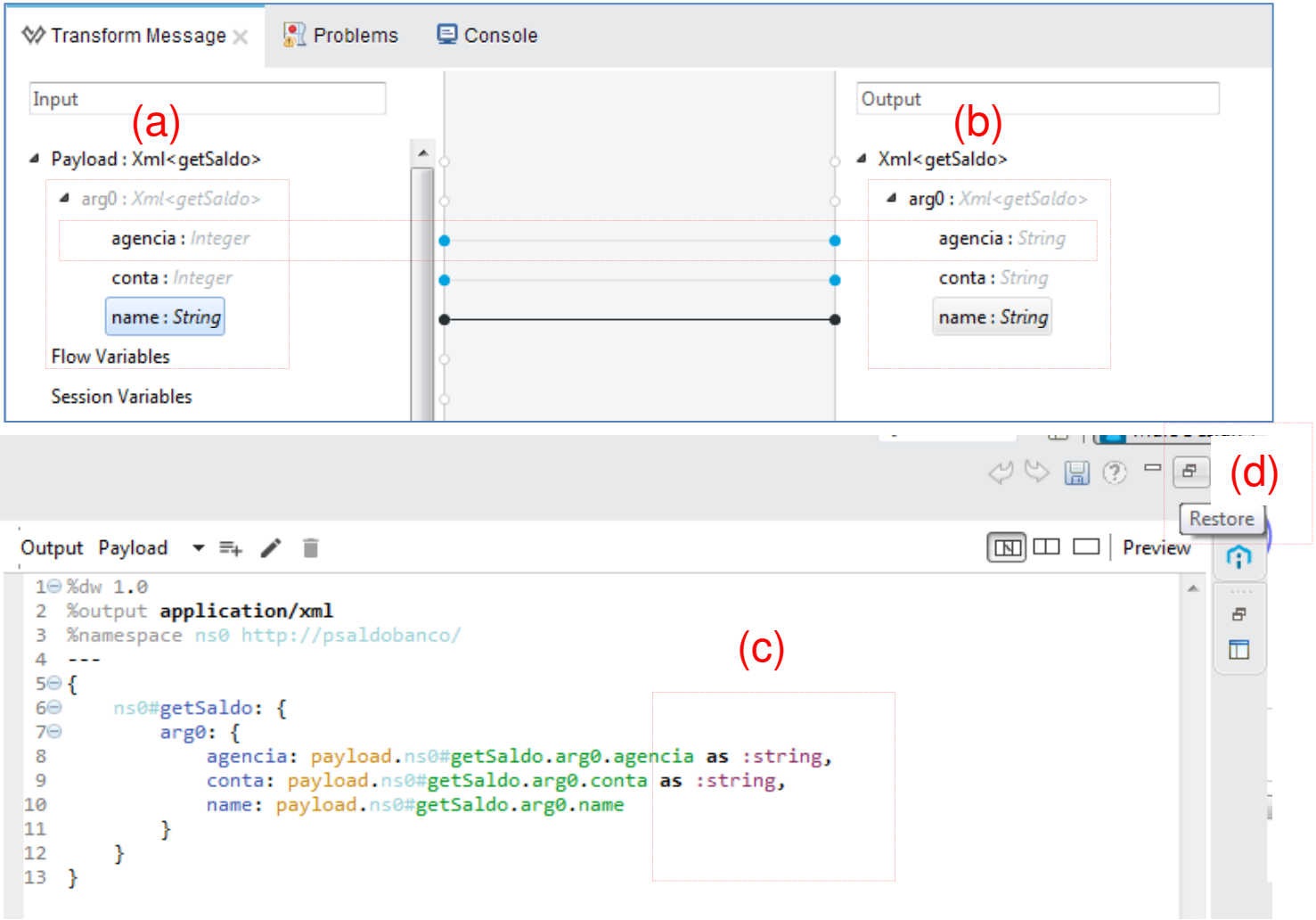
21) Maximize a tela. Clique em (a).

(a)



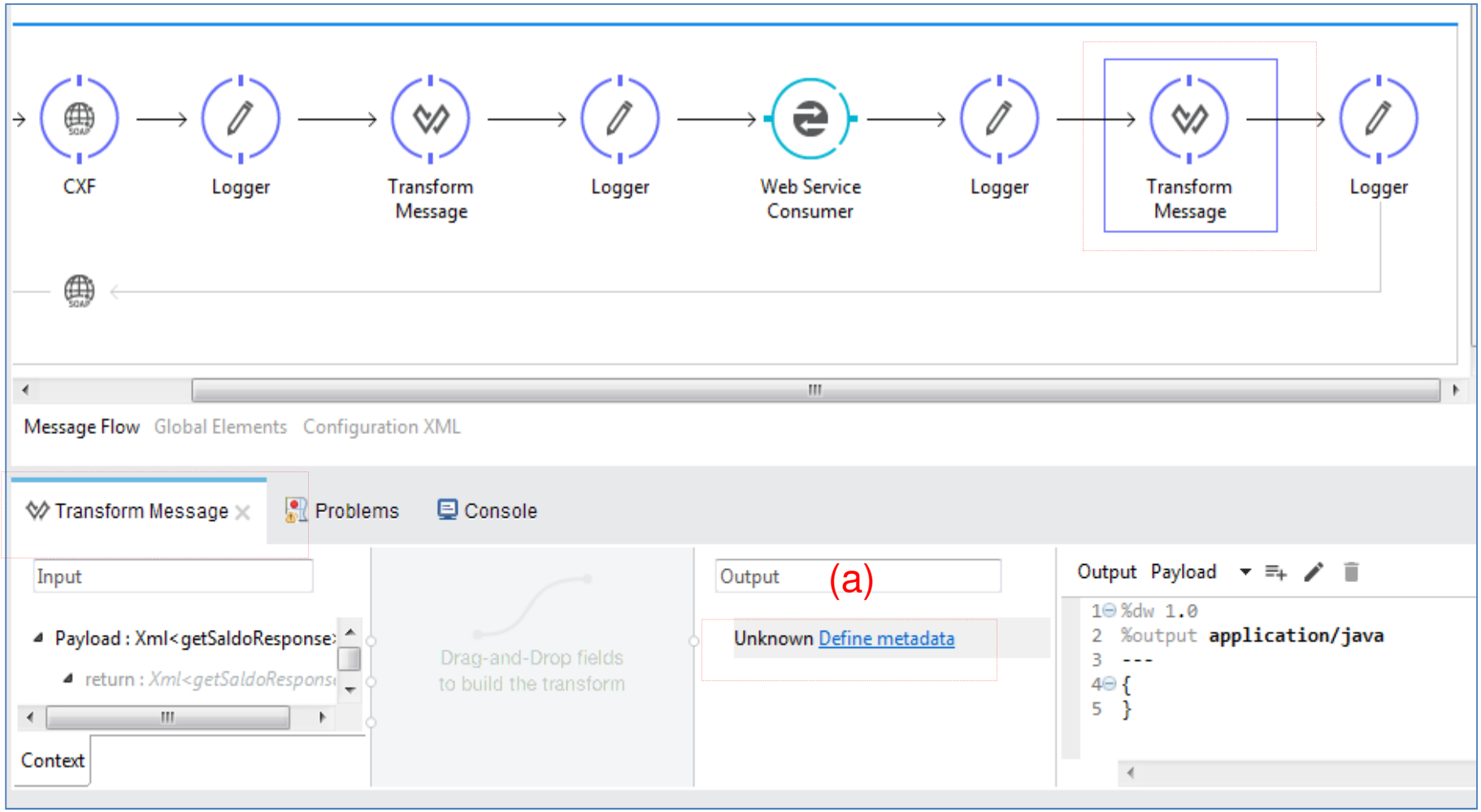
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

22) Clique em cada campo do lado (a) e arraste-o e solte-o em cima do campo do lado (b). Por exemplo, clique em “agencia: Integer” arraste-o até “agencia: String” e solte-o. Faça para os demais. Reparem que o Mule já faz a conversão automática (c). Depois restaure a tela (d) e salve o seu trabalho.



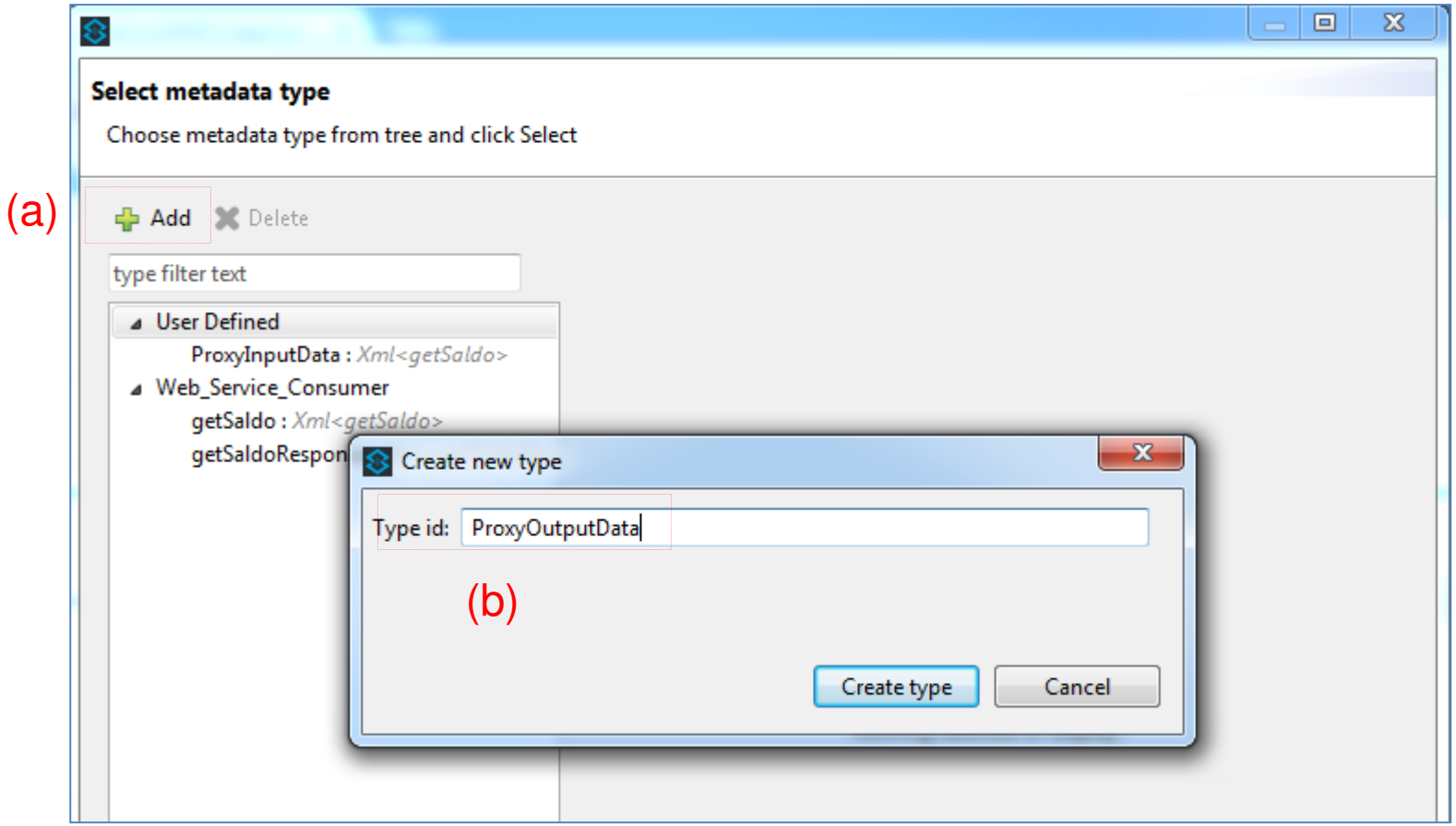
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

23) Clique no segundo componente “Transform Message”. Vamos definir o metadado de saída (que vem do webservice para o proxy). Clique em (a).



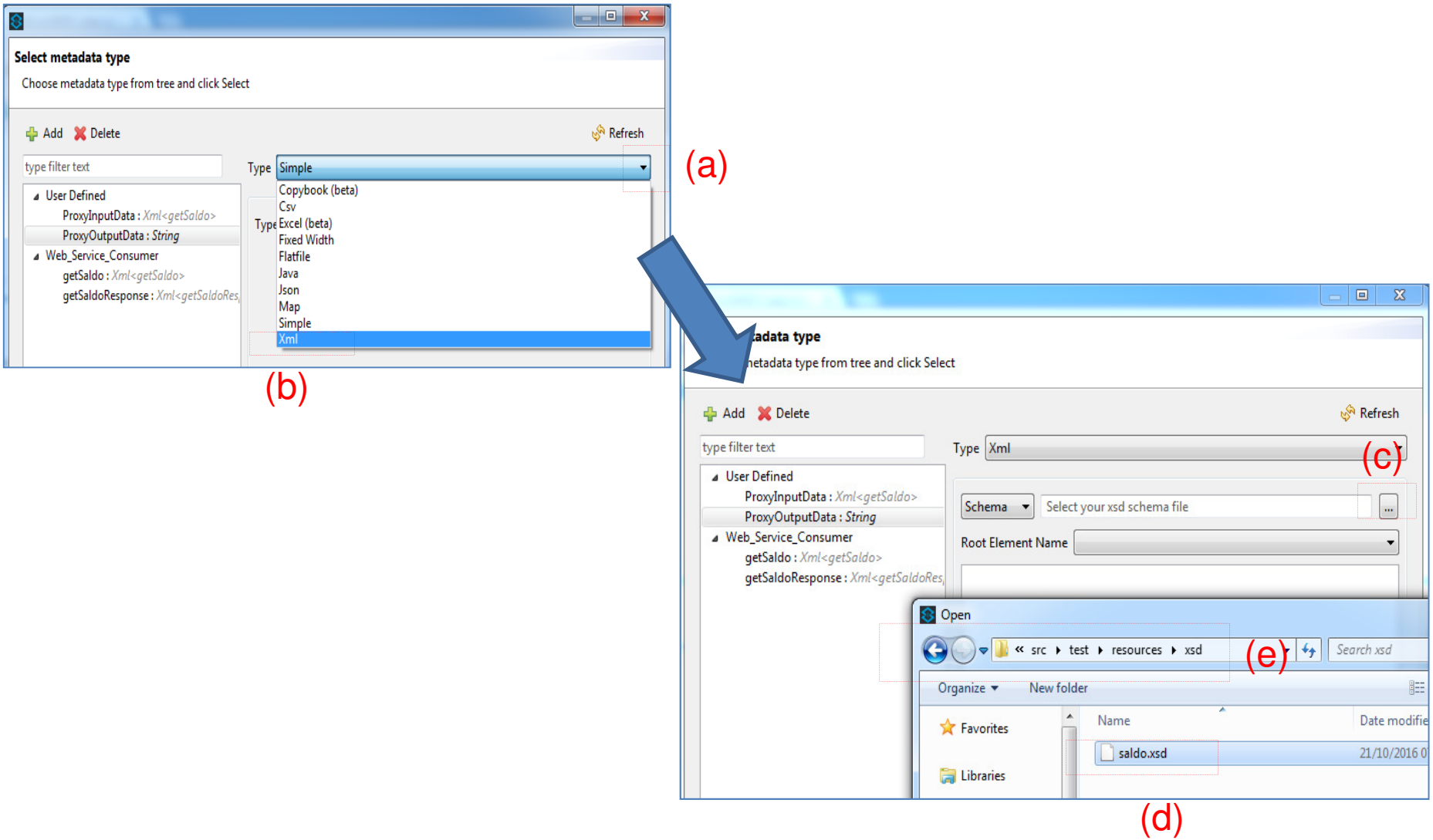
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

24) Clique no “+ verde” (a) para adicionar um novo metadado. Depois coloque o nome de “ProxyOutputData” (b).



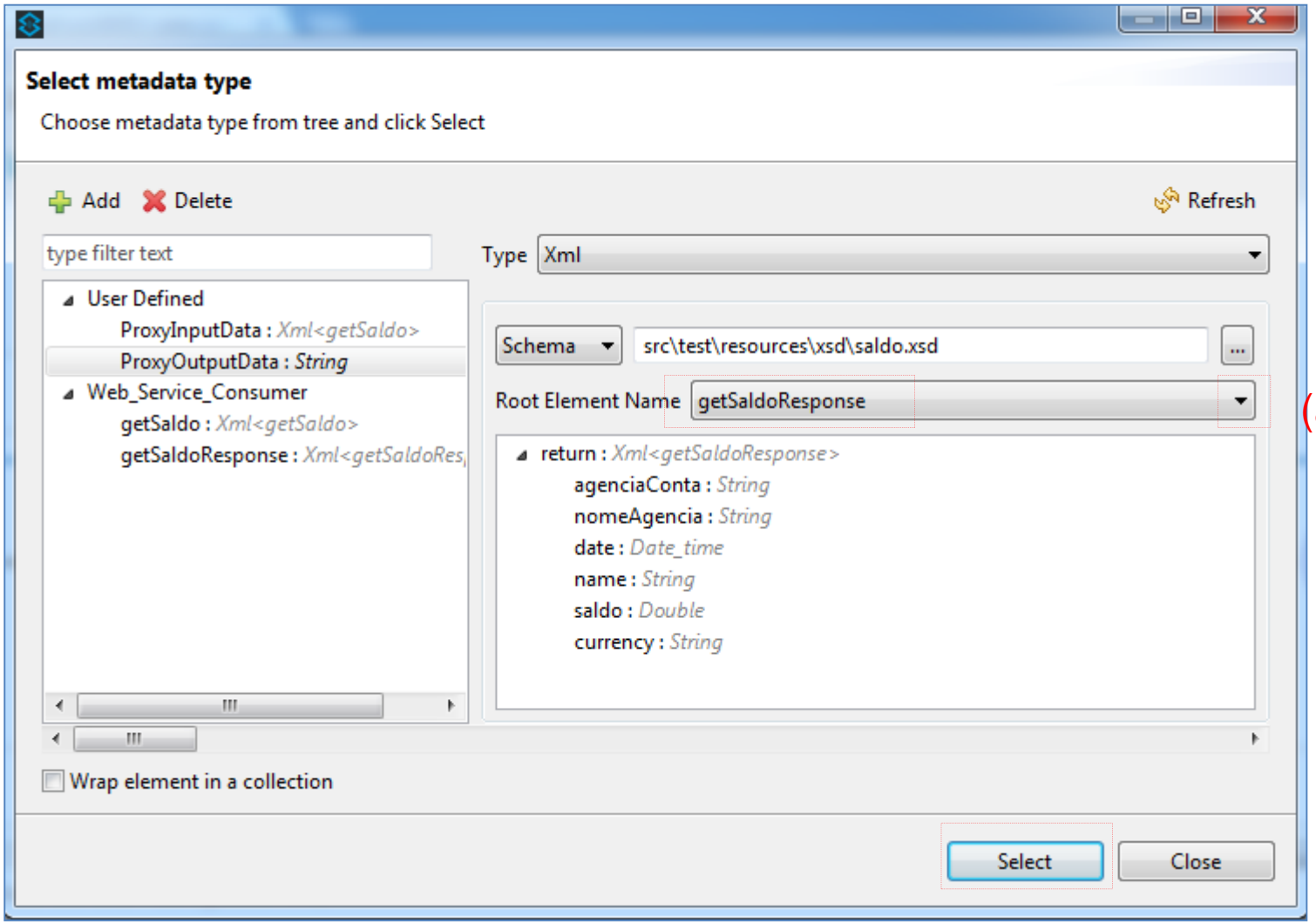
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

25) Clique em (a) e escolha “Xml” (b). Depois em (c) e escolha o arquivo “saldo.xsd” (d) que está na pasta “src/test/resources/xsd” (e).



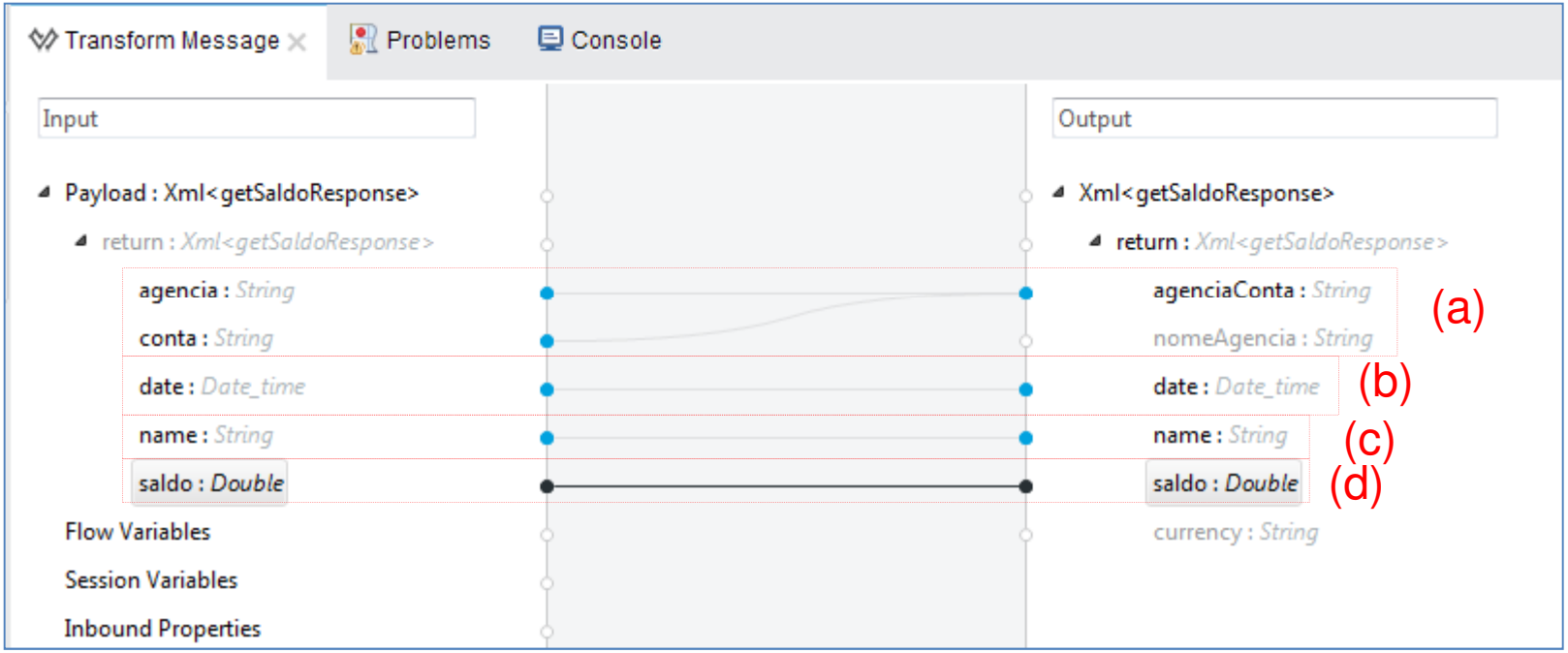
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

26) Clique em (a) e depois selecione “getSaldoResponse”. Depois clique no botão “Select”.



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

27) Maximize a tela e comece a trabalhar. Primeiramente associe os seguintes campos: “agencia” e “conta” com “agenciaConta” (a), “date” com “date” (b), “name” com “name” (c) e “saldo” com “saldo” (d) conforme figura abaixo.



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

28) Depois faça a alteração:

•agenciaConta: payload.ns0#getSaldoResponse.return.agencia ++ '/' ++ payload.ns0#getSaldoResponse.return.conta,
•nomeAgencia: 'Agencia Rua do Sol',
•date: payload.ns0#getSaldoResponse.return.date,
•name: payload.ns0#getSaldoResponse.return.name,
•saldo: payload.ns0#getSaldoResponse.return.saldo,
•currency: 'Reais'

Output Payload

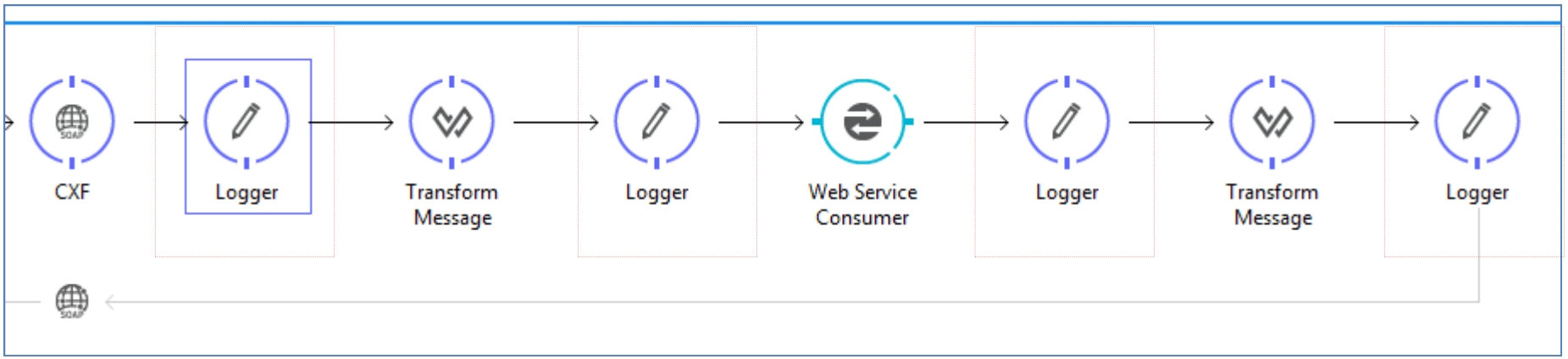
```
%dw 1.0
%output application/xml
%namespace ns0 http://psaldobanco/
---
{
    ns0#getSaldoResponse: {
        return: {
            agenciaConta: payload.ns0#getSaldoResponse.return.agencia
++ payload.ns0#getSaldoResponse.return.conta,
            date: payload.ns0#getSaldoResponse.return.date,
            name: payload.ns0#getSaldoResponse.return.name,
            saldo: payload.ns0#getSaldoResponse.return.saldo
        }
    }
}
```

Output Payload

```
%dw 1.0
%output application/xml
%namespace ns0 http://psaldobanco/
---
{
    ns0#getSaldoResponse: {
        return: {
            agenciaConta: payload.ns0#getSaldoResponse.return.agencia
++ '/' ++ payload.ns0#getSaldoResponse.return.conta,
            nomeAgencia: 'Agencia Rua do Sol',
            date: payload.ns0#getSaldoResponse.return.date,
            name: payload.ns0#getSaldoResponse.return.name,
            saldo: payload.ns0#getSaldoResponse.return.saldo,
            currency: 'Reais'
        }
    }
}
```

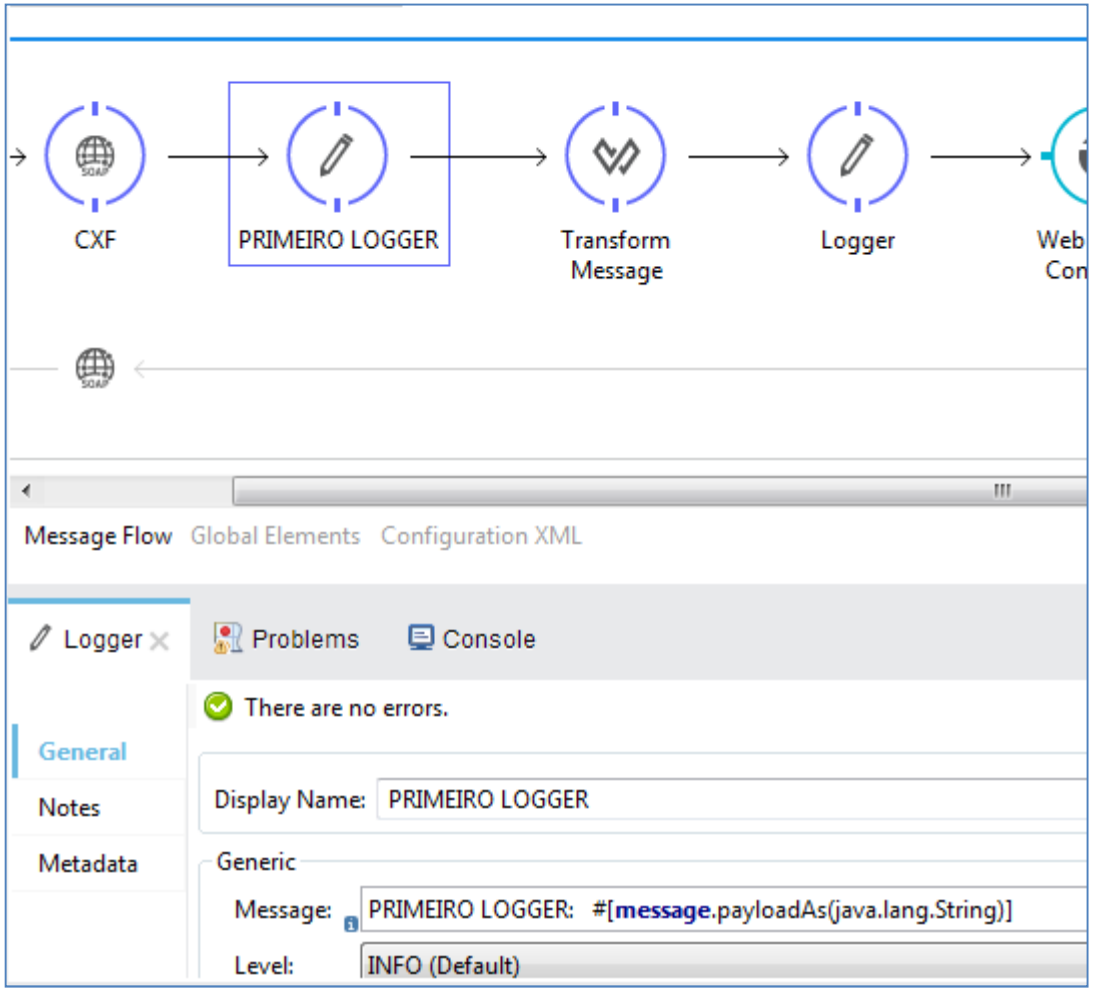
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

29) Restaure a tela e salve o seu trabalho. Agora vamos trabalhar com os componentes “Logger”.



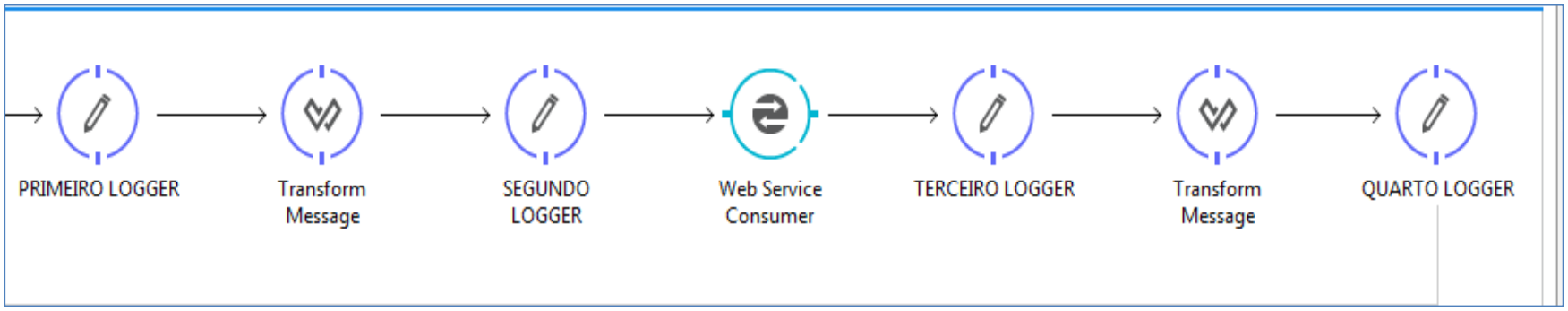
Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

30) Clique no primeiro componente “Logger”e insira as seguintes informações:
Display Name: **PRIMEIRO LOGGER**
Message: **PRIMEIRO LOGGER: #[message.payloadAs(java.lang.String)]**



Criação de um serviço “Proxy” com transformação de dados para o serviço de “Saldo”

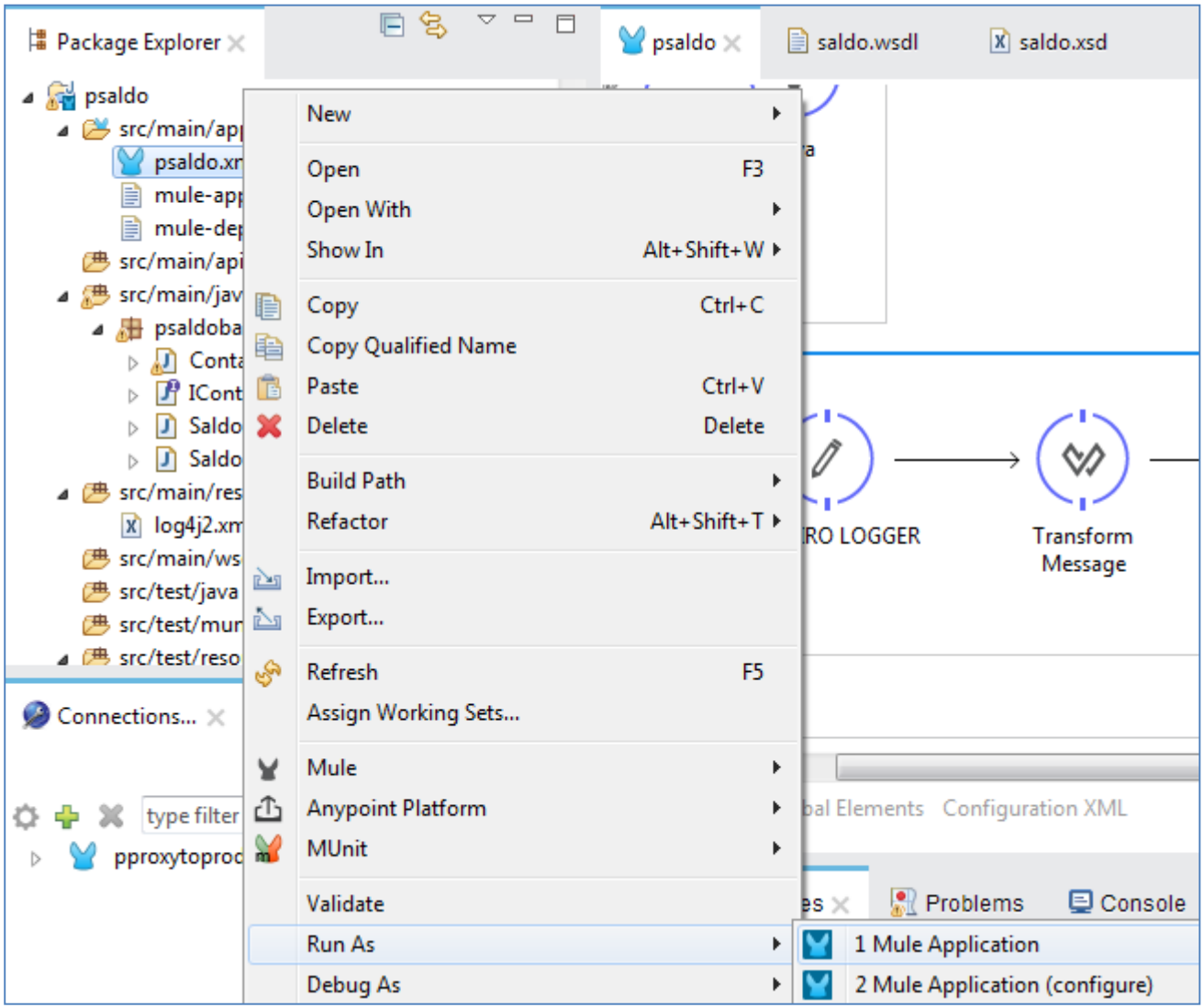
31) Repita o procedimento para os demais. Salve seu trabalho. Está pronto seu serviço de “Proxy”.



Testando o serviço “Proxy” no Mule Runtime e SoapUI

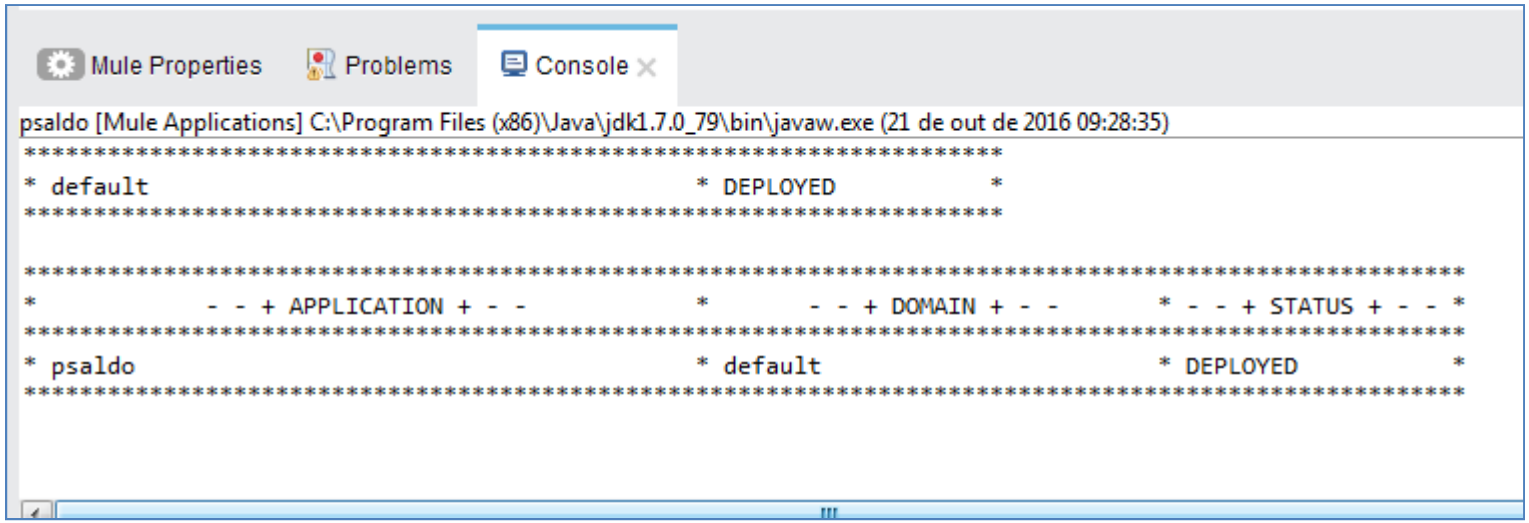
Testando o serviço “Proxy” no Mule Runtime e SoapUI

32)Clique com o botão direito em “psaldo.xml” e depois “Run As”->”Mule Application”.



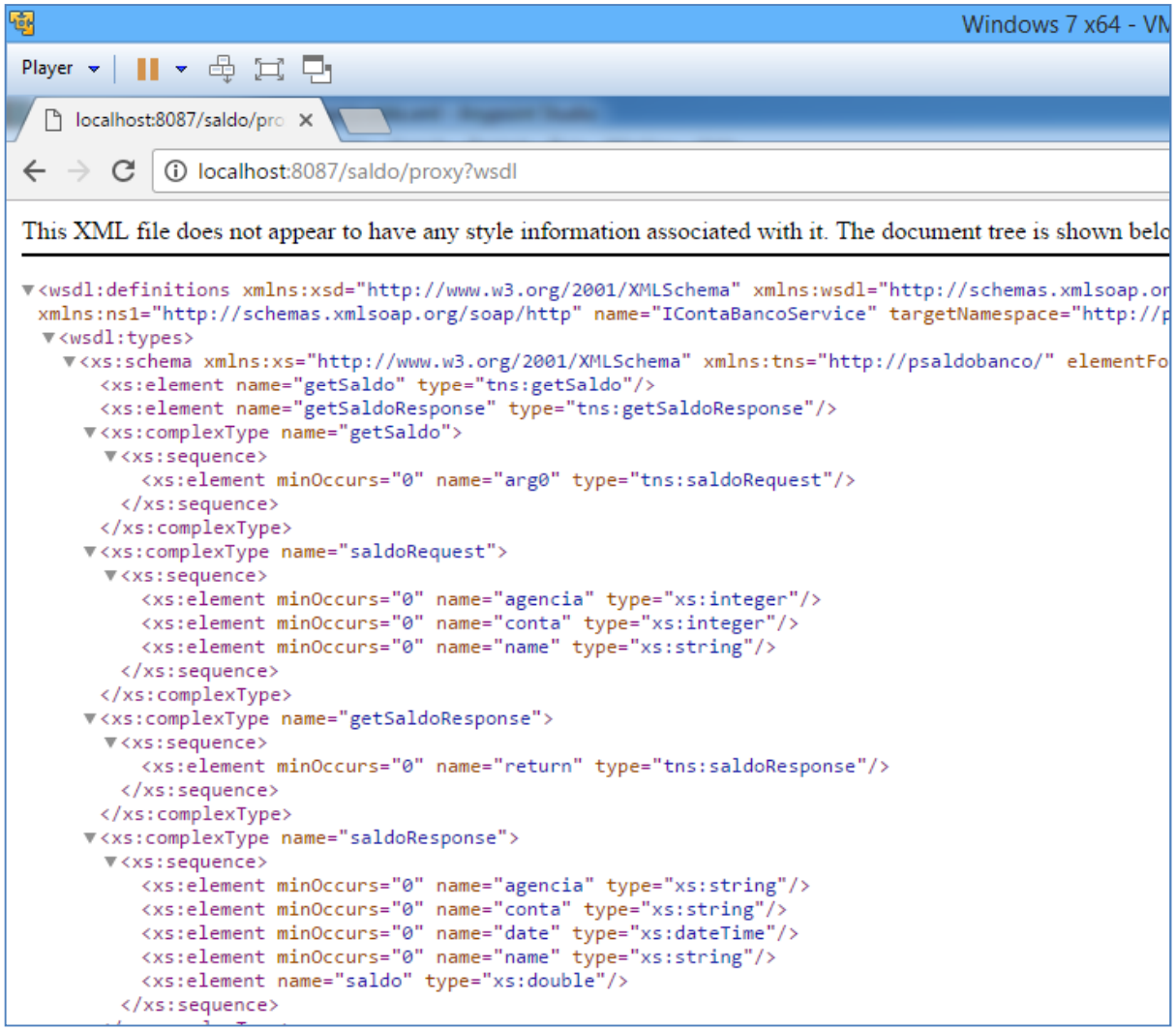
Testando o serviço “Proxy” no Mule Runtime e SoapUI

33) Na tela “Console”, abaixo, o “runtime” do Mule executará e sua aplicação será implantada “Deployed”.



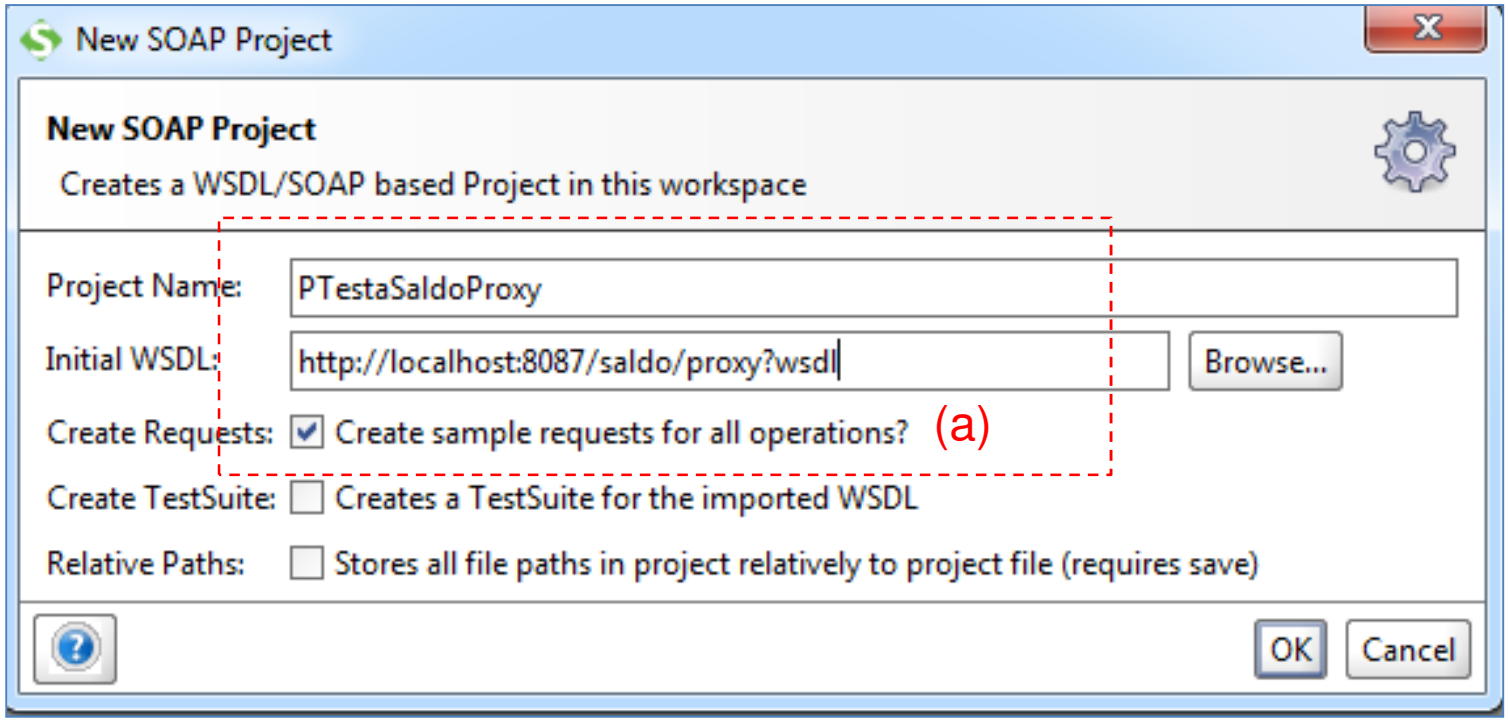
Testando o serviço “Proxy” no Mule Runtime e SoapUI

34) Abra uma janela do “Browser” e coloque o endereço: “**http://localhost:8087/saldo/proxy?wsdl**”.



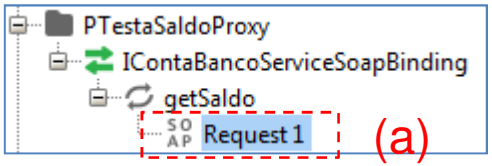
Testando o serviço “Proxy” no Mule Runtime e SoapUI

35) Agora abra o SoapUI e crie um novo projeto chamado “PTestaSaldoProxy”. Copie o endereço do wsdl para o campo “Initial WSDL” e marque na opção de criação de requisições automáticas (a) conforme figura abaixo e depois clique “OK”.



Testando o serviço “Proxy” no Mule Runtime e SoapUI

36) Abra o request (a) , preencha os campos (b) e execute o teste (c).



(c)

Request 1

http://localhost:8087/saldo/proxy

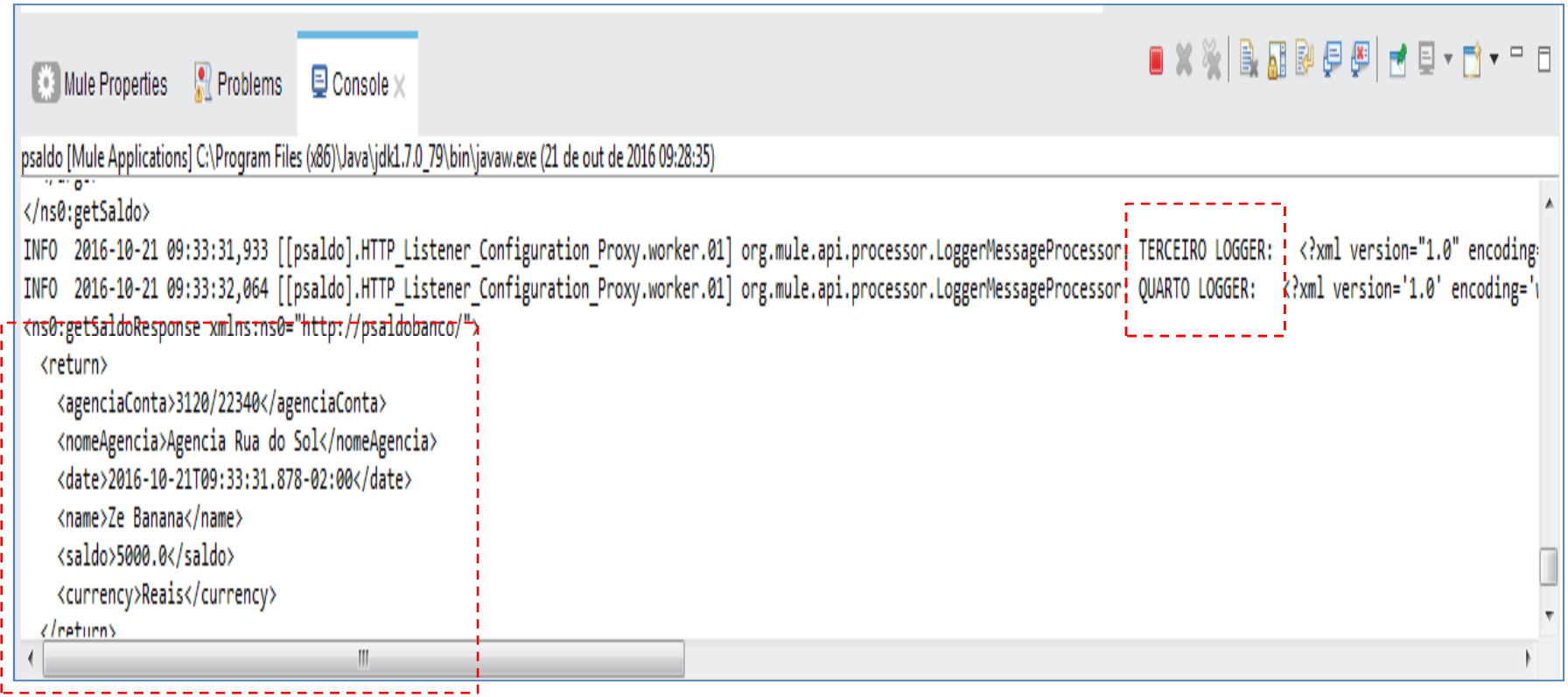
```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope
   <soapenv:Header/>
   <soapenv:Body>
      <psal:getSaldo>
         <!--Optional:-->
         <arg0>
            <!--Optional:-->
            <agencia>3120</agencia>
            <!--Optional:-->
            <conta>22340</conta>
            <!--Optional:-->
            <name>Ze Banana</name>
         </arg0>
      </psal:getSaldo>
   </soapenv:Body>
</soapenv:Envelope>
```

(b)

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
      <ns0:getSaldoResponse xmlns:ns0="http://psaldobanco/">
         <return>
            <agenciaConta>3120/22340</agenciaConta>
            <nomeAgencia>Agencia Rua do Sol</nomeAgencia>
            <date>2016-10-21T09:33:31.878-02:00</date>
            <name>Ze Banana</name>
            <saldo>5000.0</saldo>
            <currency>Reais</currency>
         </return>
      </ns0:getSaldoResponse>
   </soap:Body>
</soap:Envelope>
```

Testando o serviço “Proxy” no Mule Runtime e SoapUI

37) Volte para o console e veja o que foi “loggado”:



Source: http://www.123rf.com/photo_18476654_doodle-style-quitting-time-or-end-of-work-day-illustration-in-vector-format-includes-text-blowing-wh.html

FIAP

MBA+