

# Classificação de workflows baseados em web services

**Aluno:** Thiago Gabriel B. da Silva

**Orientador:** Prof. Dr. Renê Pegoraro

# XML

- ◉ Linguagem de marcação.
- ◉ Simples e flexível.
- ◉ Exemplo:

```
<?xml version="1.0"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

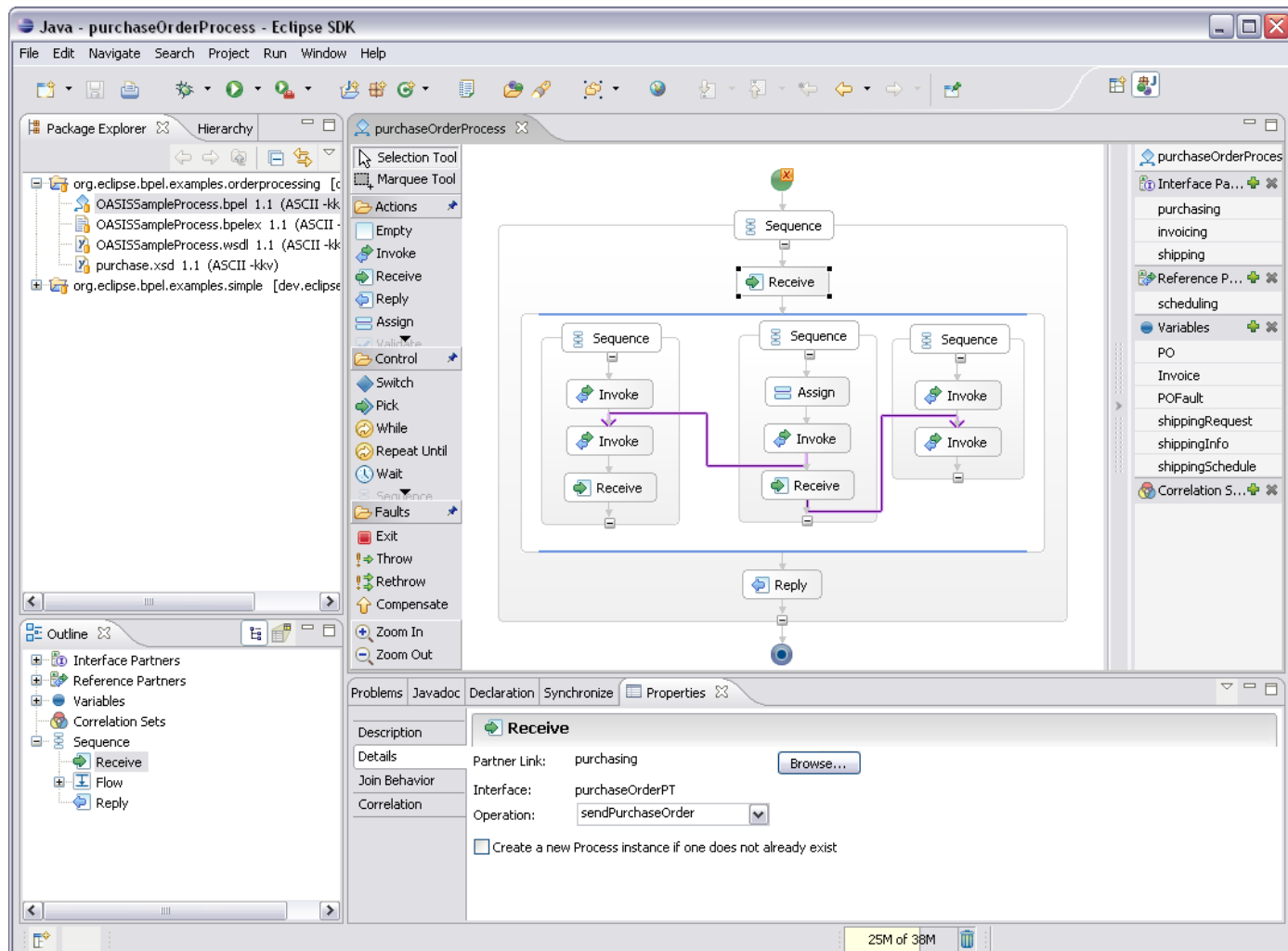
# SOAP (Simple Object Access Protocol)

- ◉ Protocolo baseado em XML para troca de informações via HTTP.
- ◉ Usado em Web Services.
- ◉ É independente de linguagem e plataforma.
- ◉ Utiliza WSDL (Web Services Description Language) para descrever o serviços, parâmetros esperados e estruturas de dados.

# BPEL (Business Process Execution Language)

- Linguagem para descrever e executar processos de negócios dentro de web services.
- Possui estruturas básicas de programação com condicionais e loops.

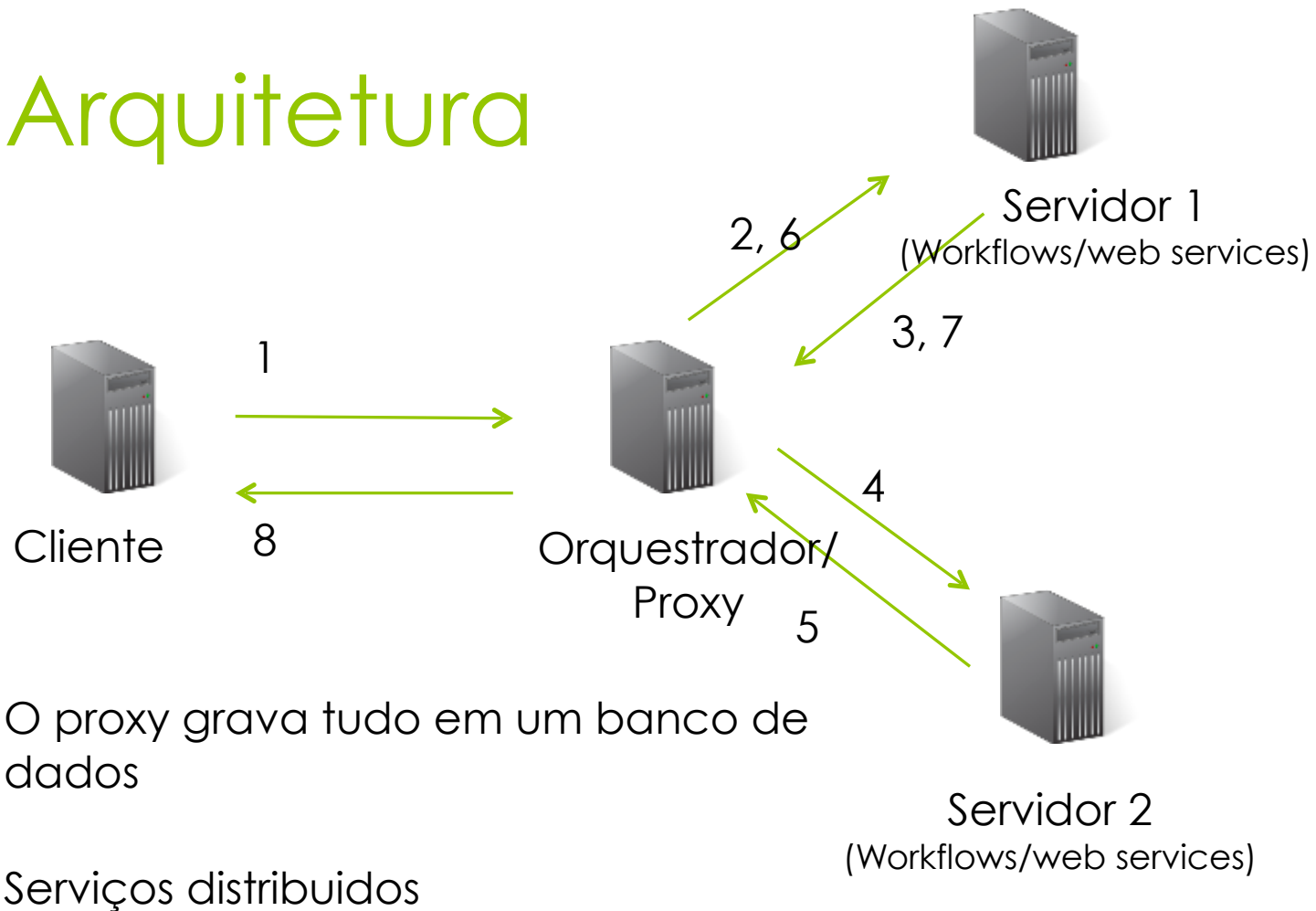
# Eclipse BPEL



# Orquestração



# Arquitetura

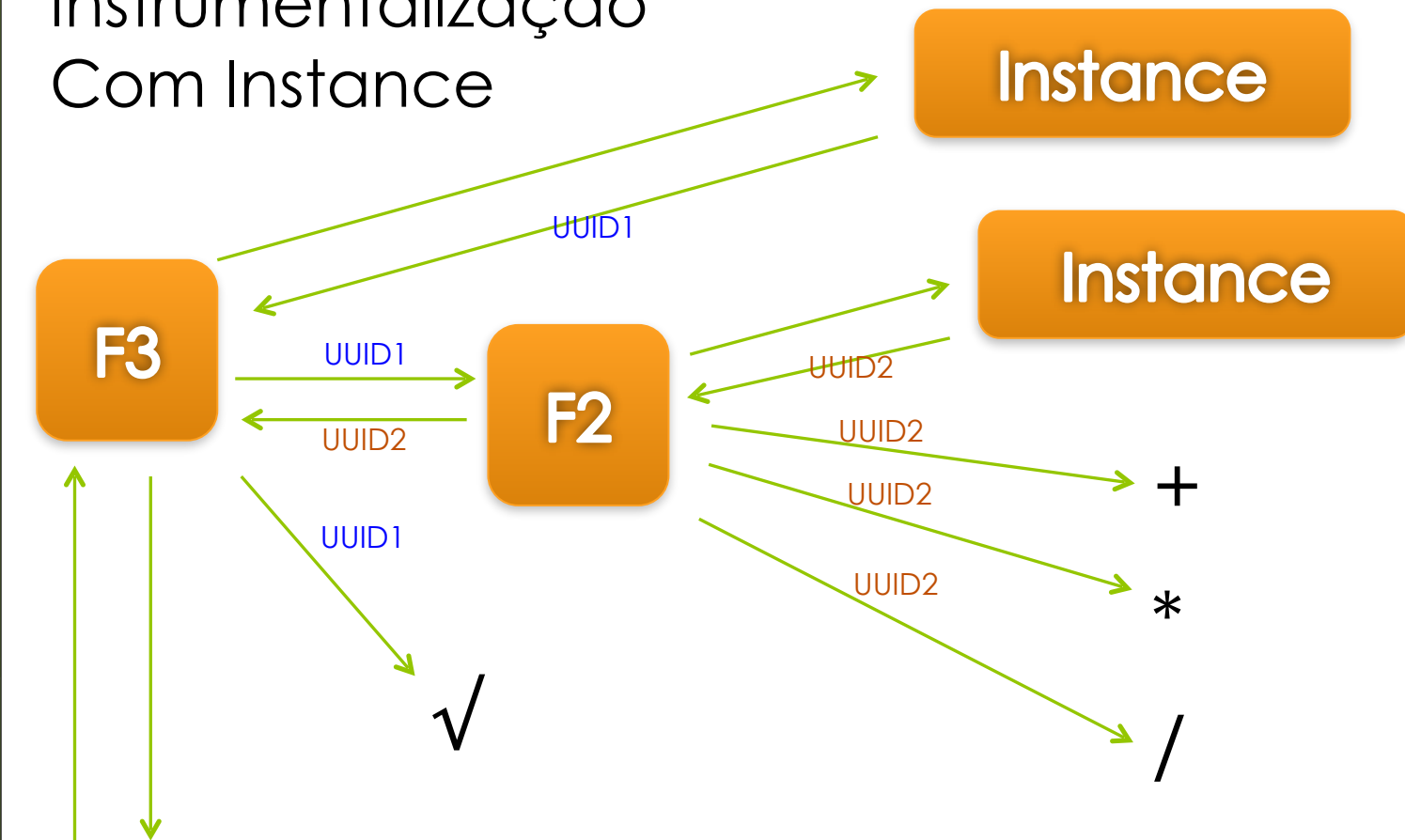


# Arquitetura

- O proxy grava tudo em um banco de dados
  - Quem faz a chamada
  - Qual serviço foi chamado
  - Qual função foi chamada
  - Hora da chamada
  - Hora da resposta



# Instrumentalização Com Instance

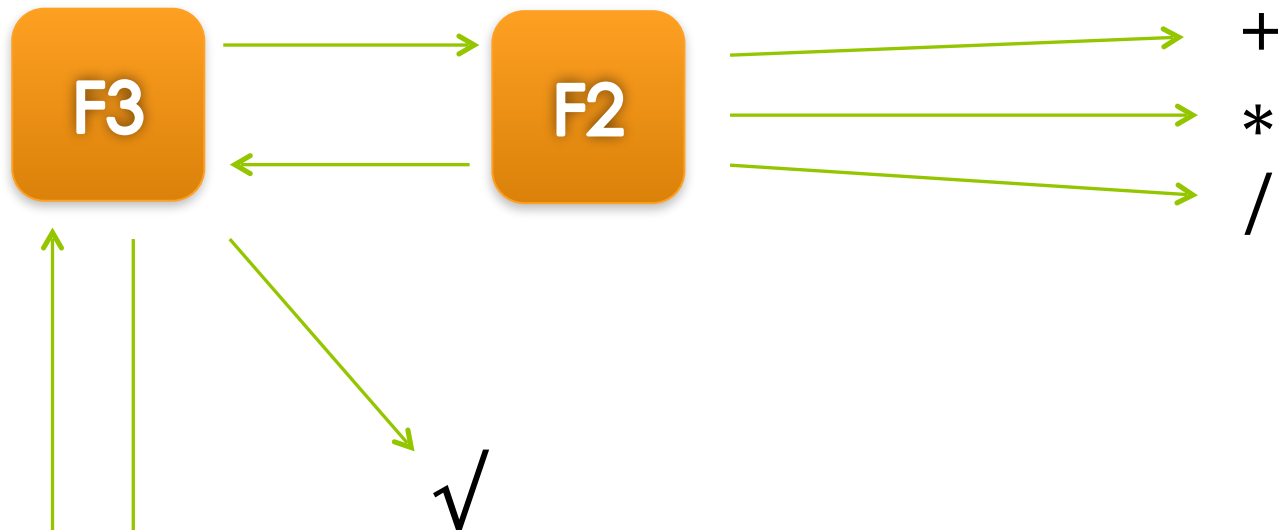


$$F2(x, y) = (x+y) / (x*y)$$

$$F3(x, y) = \sqrt{F2(x, y)}$$

$$\sqrt{(A + B) / (A * B)}$$

# Instrumentalização Sem Instance

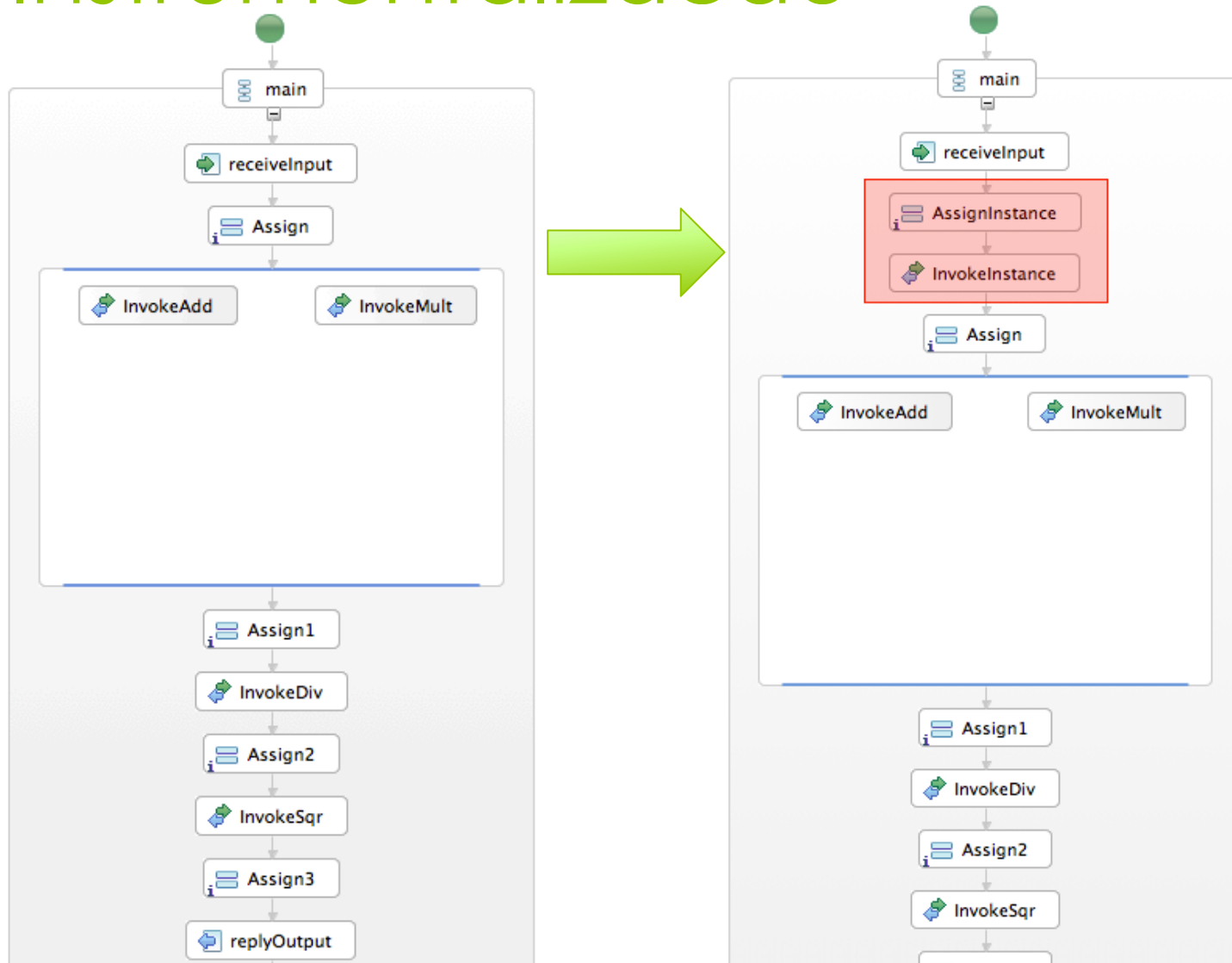


$$F2(x, y) = (x+y) / (x*y)$$

$$F3(x, y) = \sqrt{F2(x, y)}$$

$$\sqrt{(A + B) / (A * B)}$$

# Instrumentalização

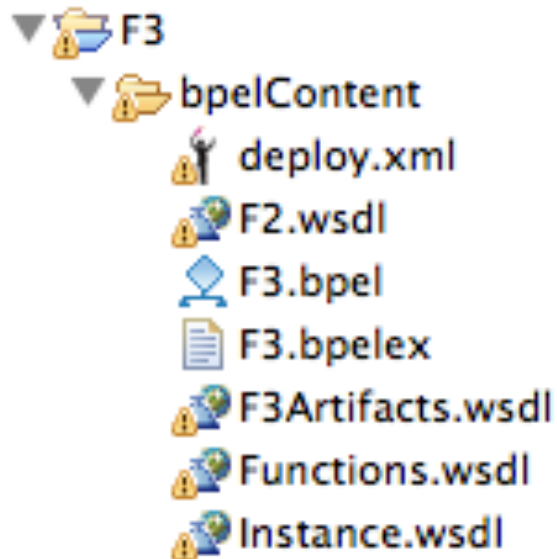


# A instrumentalização do Workflow

- Pode ser automatizado.
- Estágios:
  - Criar um novo serviço para gerar um UUID para cada instância.
  - Alterar o BPEL.
    - Chamar o webservice para gerar novo UUID (número da instância).
    - Adicionar UUID para cada chamada de web services.
    - Retornar o UUID.
  - Alterar os WSDLs.
    - Adicionar o parâmetro que carrega o UUID.

Exemplo UUID = 56ef4583-d46d-4163-b180-6d85e95a0436

# Instrumentalização



Parser XML para incluir código dentro do F3.bpel, e todos os .WSDL que não tenham sido previamente modificados.

# Instrumentalização

- XML Parser (VTD XML)
  - XPath /Persons/Person/Age

<Persons>

<Person Height="180">

<FullName>Yosi Havia</FullName>

<Age>18</Age>

</Person>

<Person Height="177">

<FullName>Yosi Cohen</FullName>

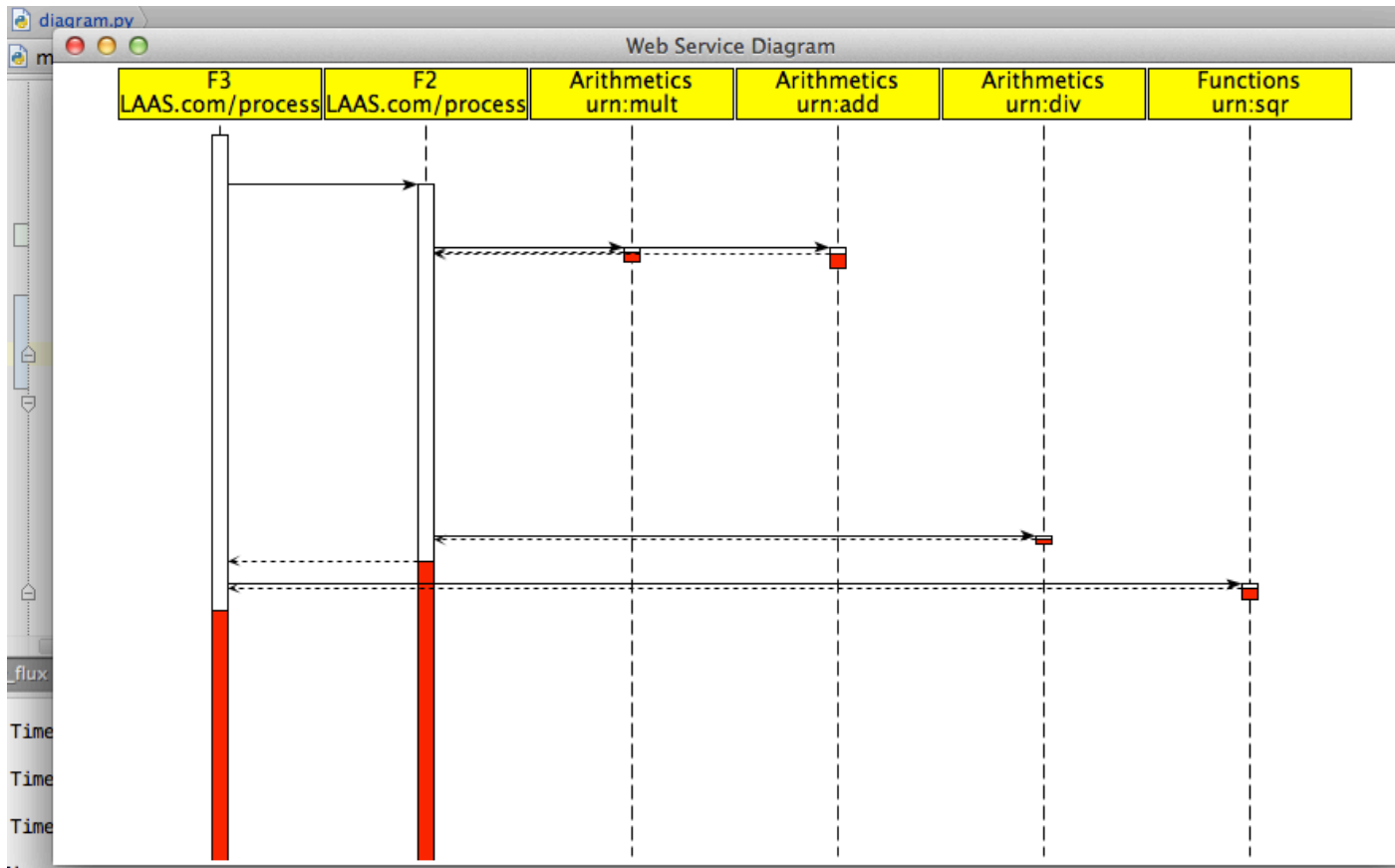
<Age>22</Age>

</Person>

</Persons>

# Gráficos

- Representação gráfica das informações como o diagrama de sequência.



# Gráficos

- Python Tkinter (Canvas) para interface gráfica.
- Dados recebidos do MySQL.
- Estrutura de grafos para percorrer os dados.



# Conclusão

- É possível criar uma ferramenta gráfica para auxiliar a análise de workflows de web services.
- Pode ser feito sem modificar os serviços de terceiros.

# Trabalhos futuros

- ◉ Estudo de outra maneira representar os dados. Média e desvio padrão podem não ser tão significativos.
- ◉ Melhorar a representação gráfica de loops e condicionais no BPEL
- ◉ Fazer com que o cliente não precise configurar o proxy para realizar a primeira chamada.

Obrigado.

# Referências

- BPEL Designer, Disponível em: <<http://www.eclipse.org/bpel>>. Acesso em 27 abr. 2012
- PEGORARO, R. et al. **A framework for monitoring and runtime recovery of web service-based applications**. Barcelona, Espagne: [s.n.],2008.
- REPP, N. et al. **A cross-layer approach to performance monitoring of web services**. In: PAUTASSO, C. et al. (Ed.). Emerging Web Services Technology. [S.l.]: Birkhauser Basel, 2007.
- SAMPAIO, C. **SOA e Web Services em Java**. Brasport, 2006. ISBN 9788574522678.
- VTD-XML, Disponível em: <<http://vtd-xml.sourceforge.net>>. Acesso em 22 jun. 2012.