

Componentização e Reúso de Software

Composição

André Santanchè

Laboratory of Information Systems - LIS

Instituto de Computação - UNICAMP

Museu Exploratório de Ciências da Unicamp

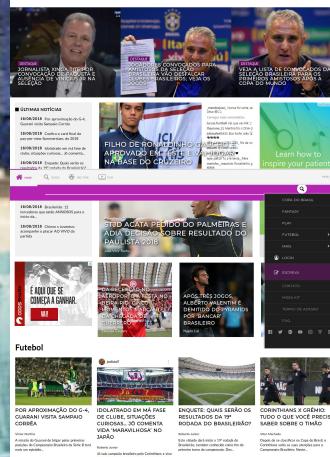
Agosto 2019

Jornalismo Colaborativo

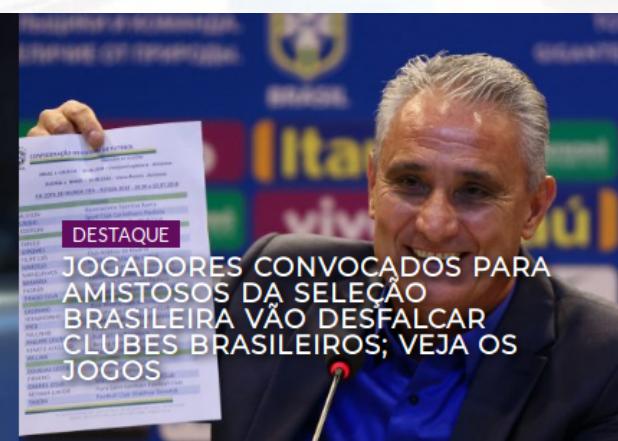
Wiki Notícias



Aproveite Mega Promo Latam - Ofertas a Partir de
R\$99!



torcedores.com



ÚLTIMAS NOTÍCIAS

18/08/2018 Por aproximação do G-4, Guarani visita Sampaio Corrêa

18/08/2018 Confira o card final do pay-per-view Summerslam, de 2018

18/08/2018 Idolatrado em má fase de clube, situações curiosas... Jô comenta...

18/08/2018 Enquete: Quais serão os resultados da 19ª rodada do Brasileirão?



mendesjoao Nunca foi sorte, se Deus ☺

Carregar mais comentários

luccas.football Vai que vai mlk ☺

thayrone_21 Mentira foi o DNA d julianaleal1611 Boa Sorte João ☺

mimodeler_1 @2_993 fils de roi elisach @Catarina413

amerasa_770

laye_sall10 Tu a le tenla_frère marinho_mauricio Ronaldinho g kkkkk



Mais notícias



HOME



BUCHA



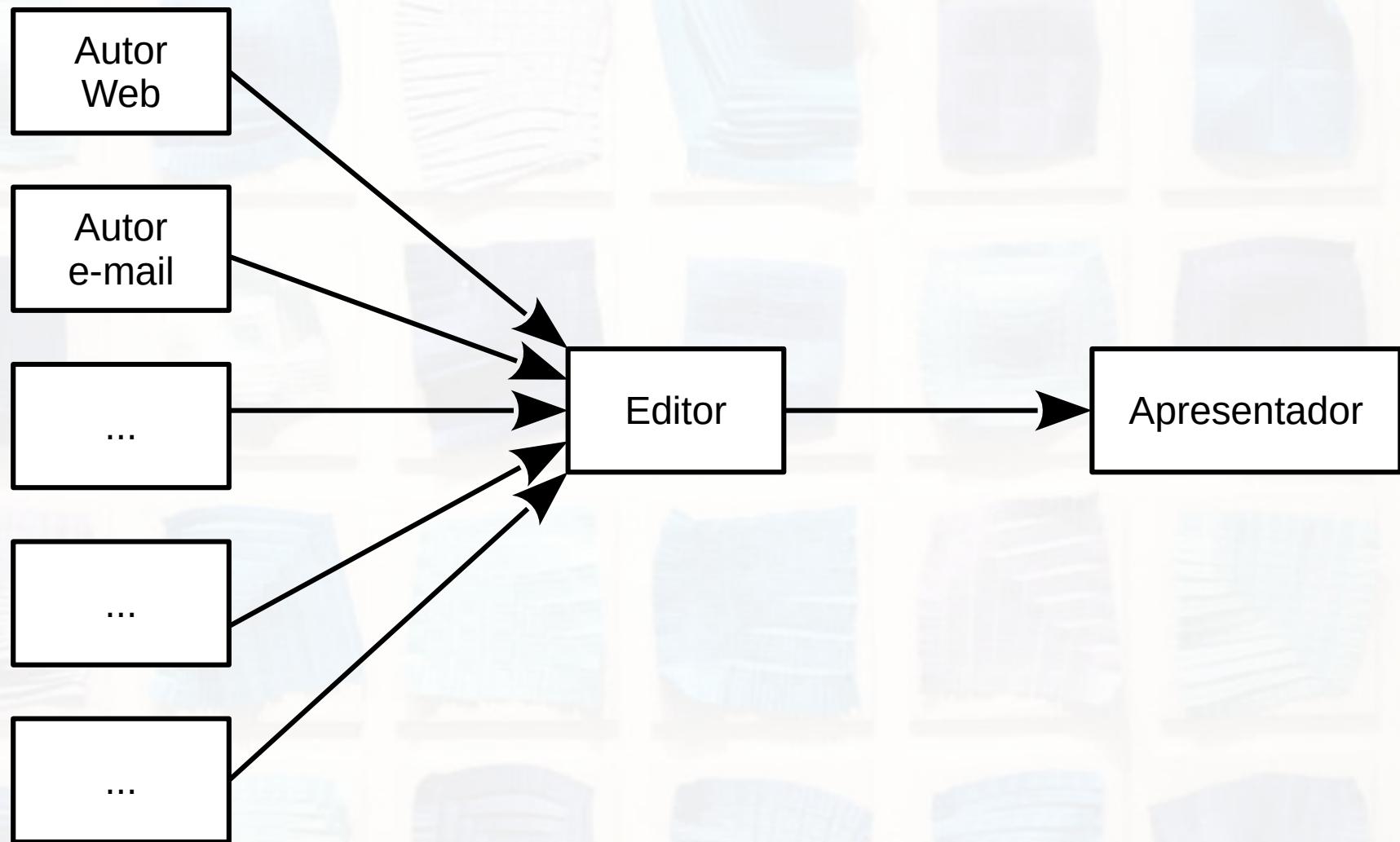
MEU TIME



PLAY

MENU

Cenário 1

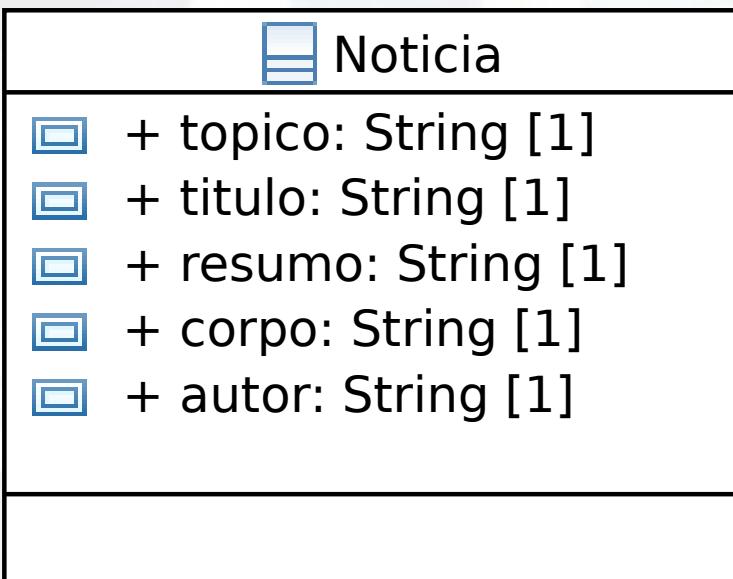


Tarefa 1

- Considere que haverá um ou mais DTOs (Data Transfer Objects) que circulará(ão) entre os componentes.
- Escreva uma classe UML para este DTO.

Tarefa 1

- Considere que haverá um ou mais DTOs (Data Transfer Objects) que circulará(ão) entre os componentes.
- Escreva uma classe UML para este DTO.

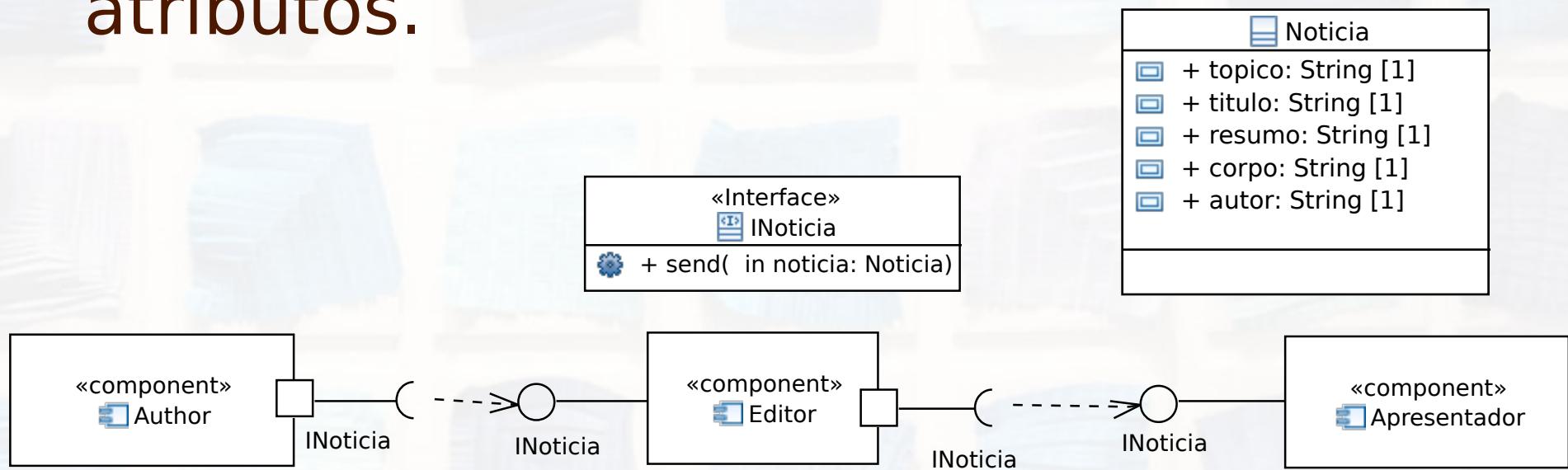


Tarefa 2

- Modele um conjunto de componentes em UML e suas respectivas interfaces, que atendam ao problema detalhado.

Tarefa 2

- Modele um conjunto de componentes em UML e suas respectivas interfaces, que atendam ao problema detalhado.
- Não é necessário especificar o tipo dos atributos.

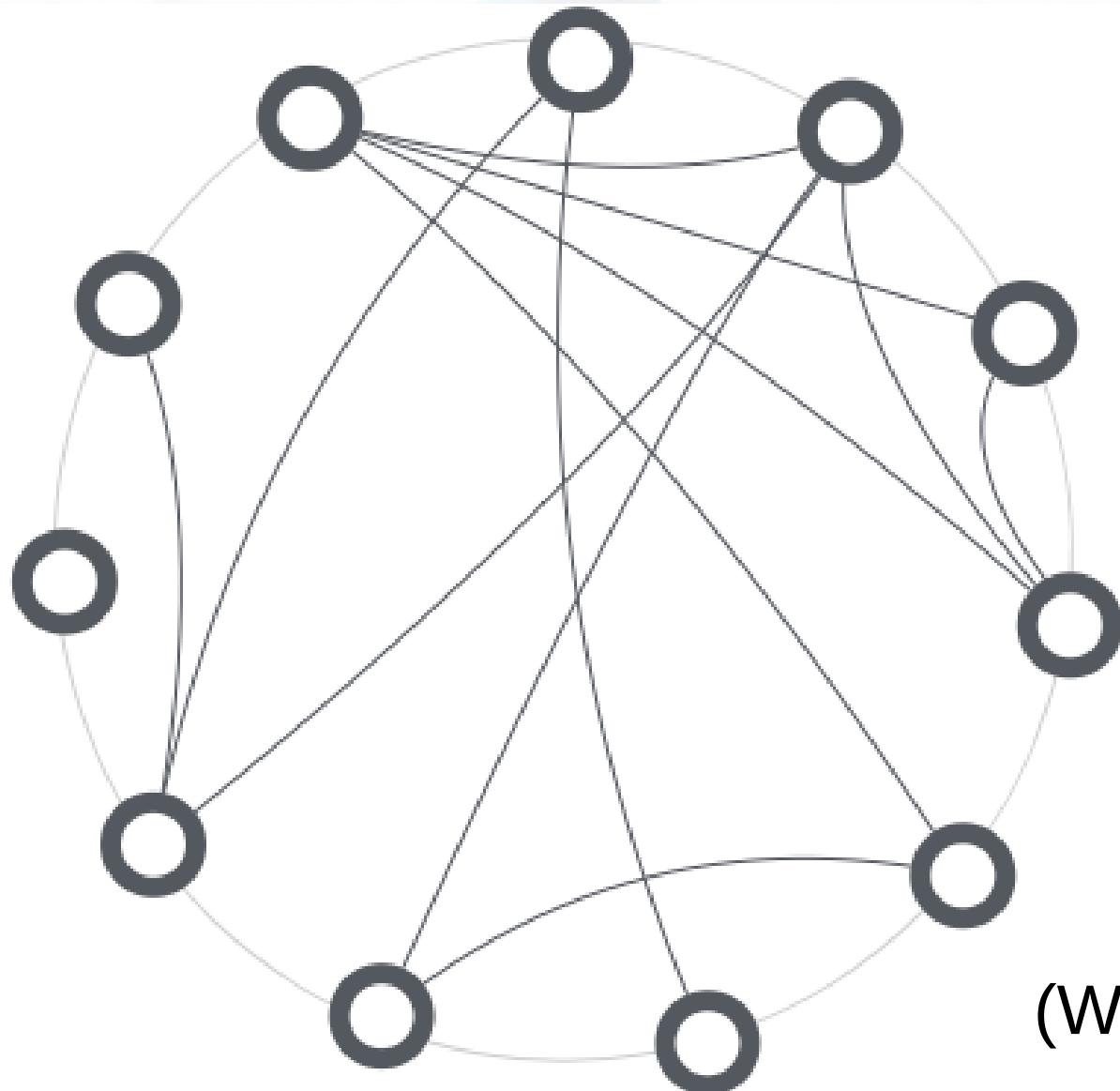


Como compor e coordenar os componentes?

Orquestraçāo



Orquestração



(Wolf, 2018)

Arquitetura na Prática

Compiere®

Compiere Workflows

■ General Workflow

- Provides guidance and step-by-step instructions for achieving a task. Examples: Setup Wizards or Month End procedures. A user starts them from the menu.

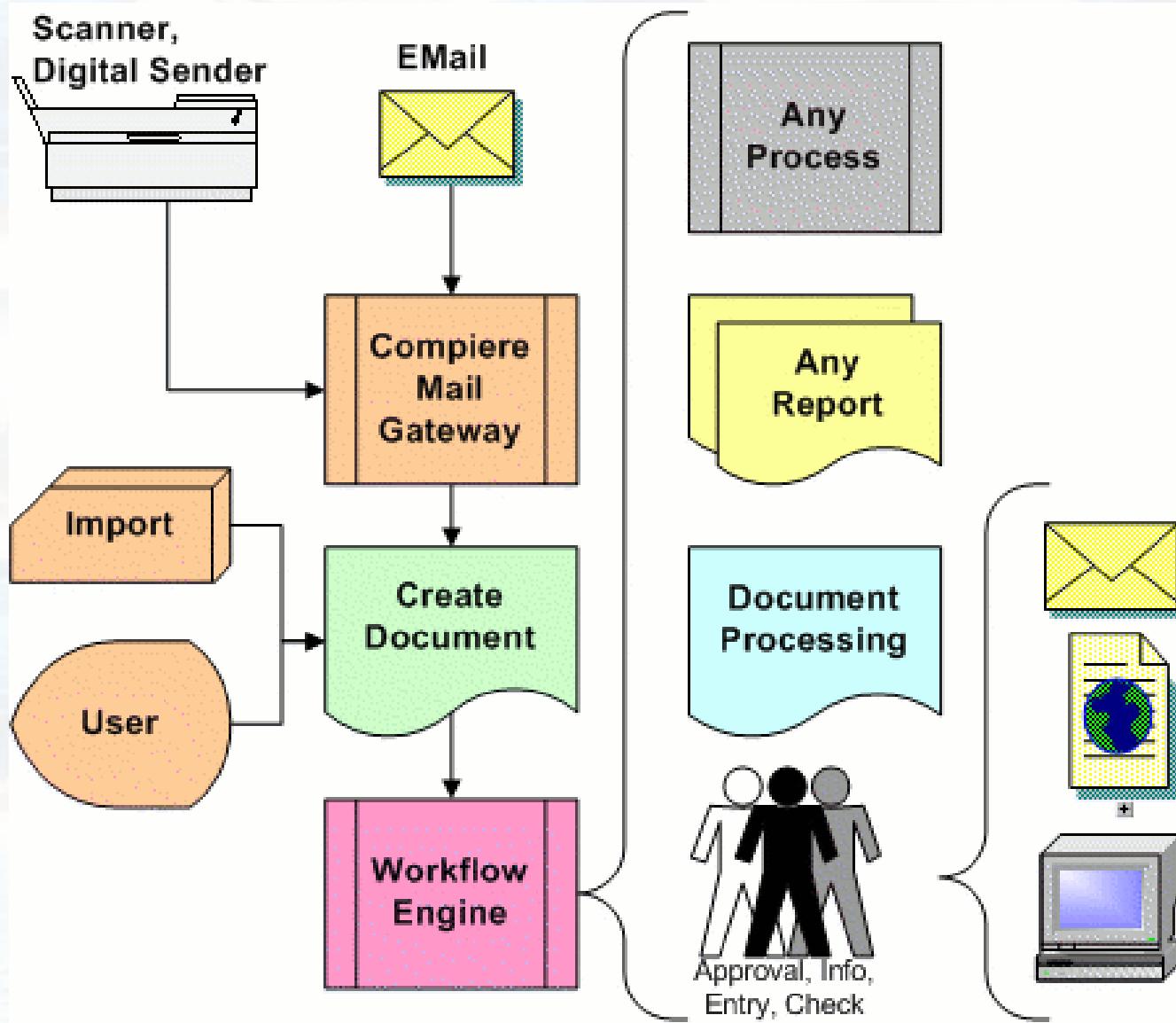
■ Document Process Workflow

- Started when processing any document. You would extend these workflow type for approval situations. Example: Special approval for orders over a certain amount.

■ Document Value Workflow

- The workflow is automatically started when any entity fulfills a user defined condition. Example: Start credit approval for a new Business Partner.

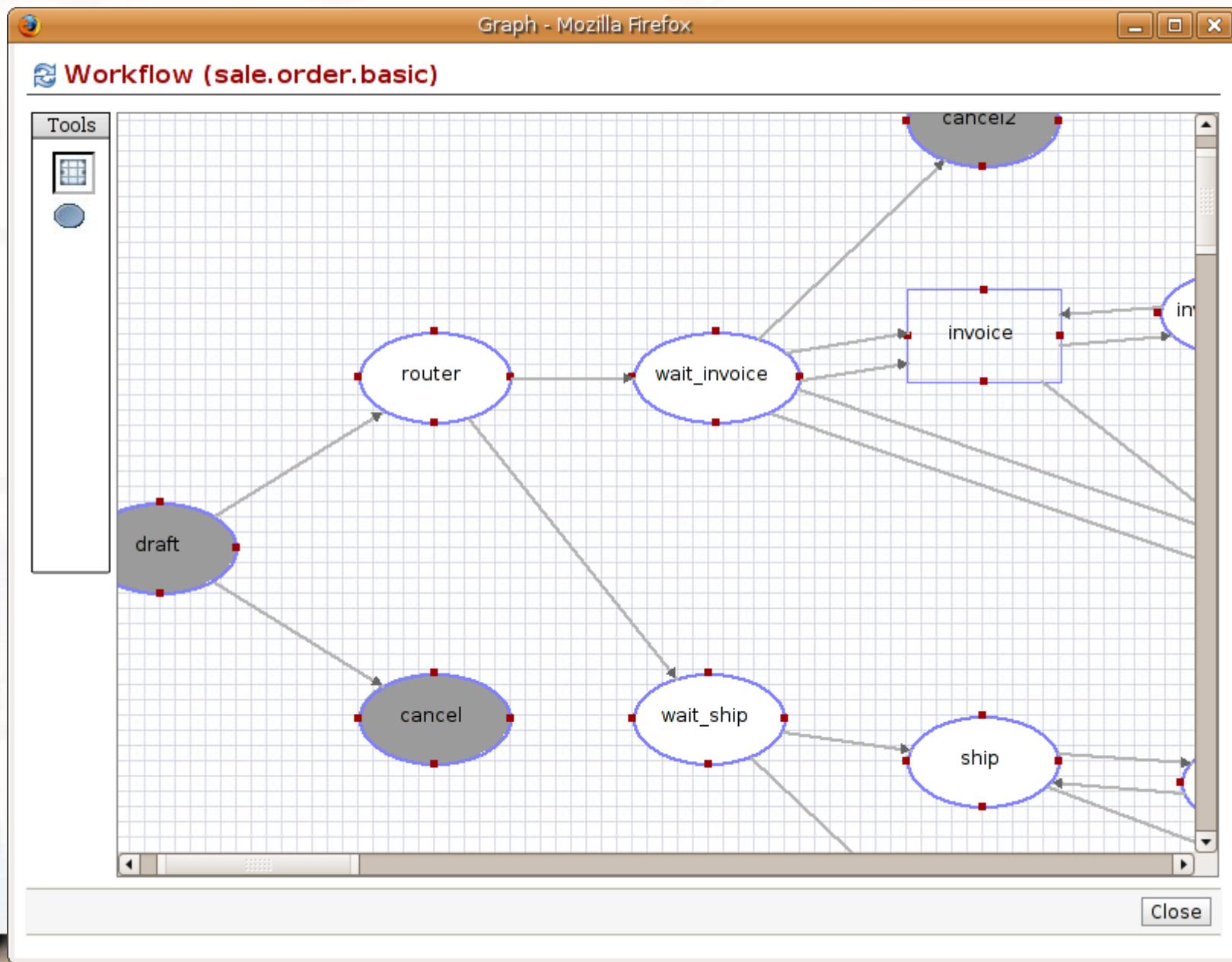
Compiere Workflows



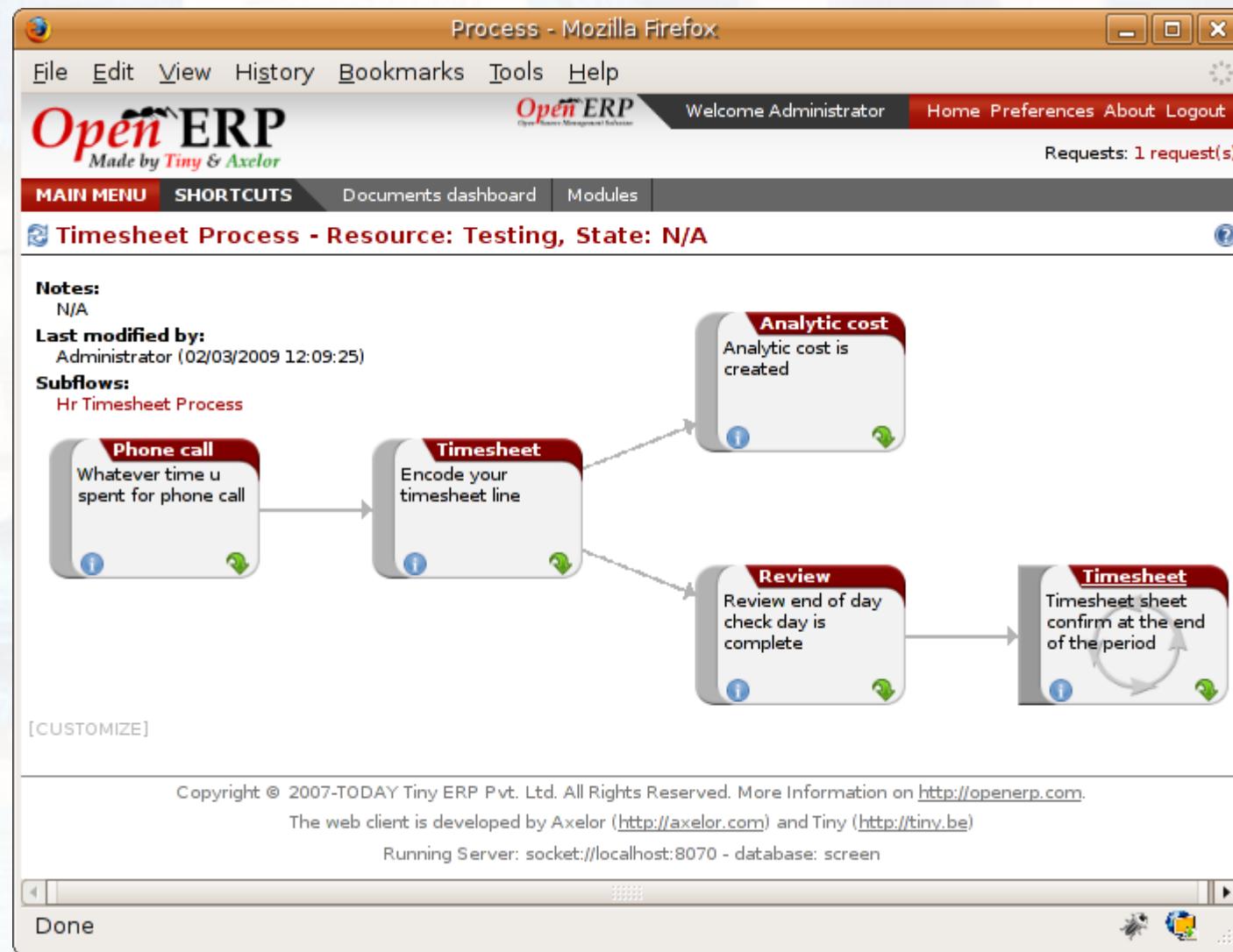
Arquitetura na Prática

Open ERP
OPEN SOURCE MANAGEMENT SOLUTION

Open ERP Workflows



Open ERP Workflows



Open ERP Workflows



OpenERP
Open Source Management System

Welcome Administrator

Home Preferences About Logout

Requests: 1 request(s)

MAIN MENU SHORTCUTS Documents dashboard Modules

Sales Process - Resource: SO005, State: Manual In Progress



Notes:

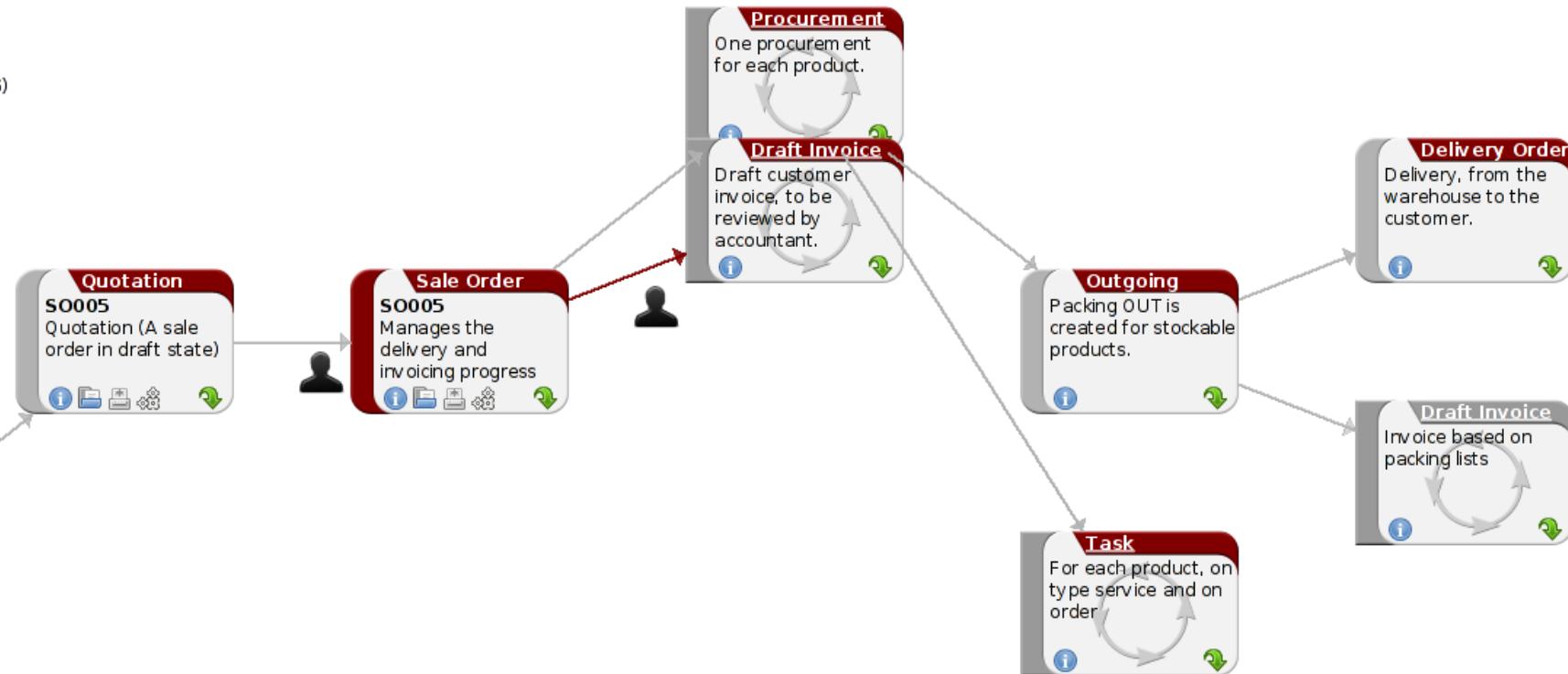
N/A

Last modified by:

Administrator (02/03/2009 10:35:56)

Subflows:

Procurement Process
Customer Invoice Process
Tasks Process
Contract Process



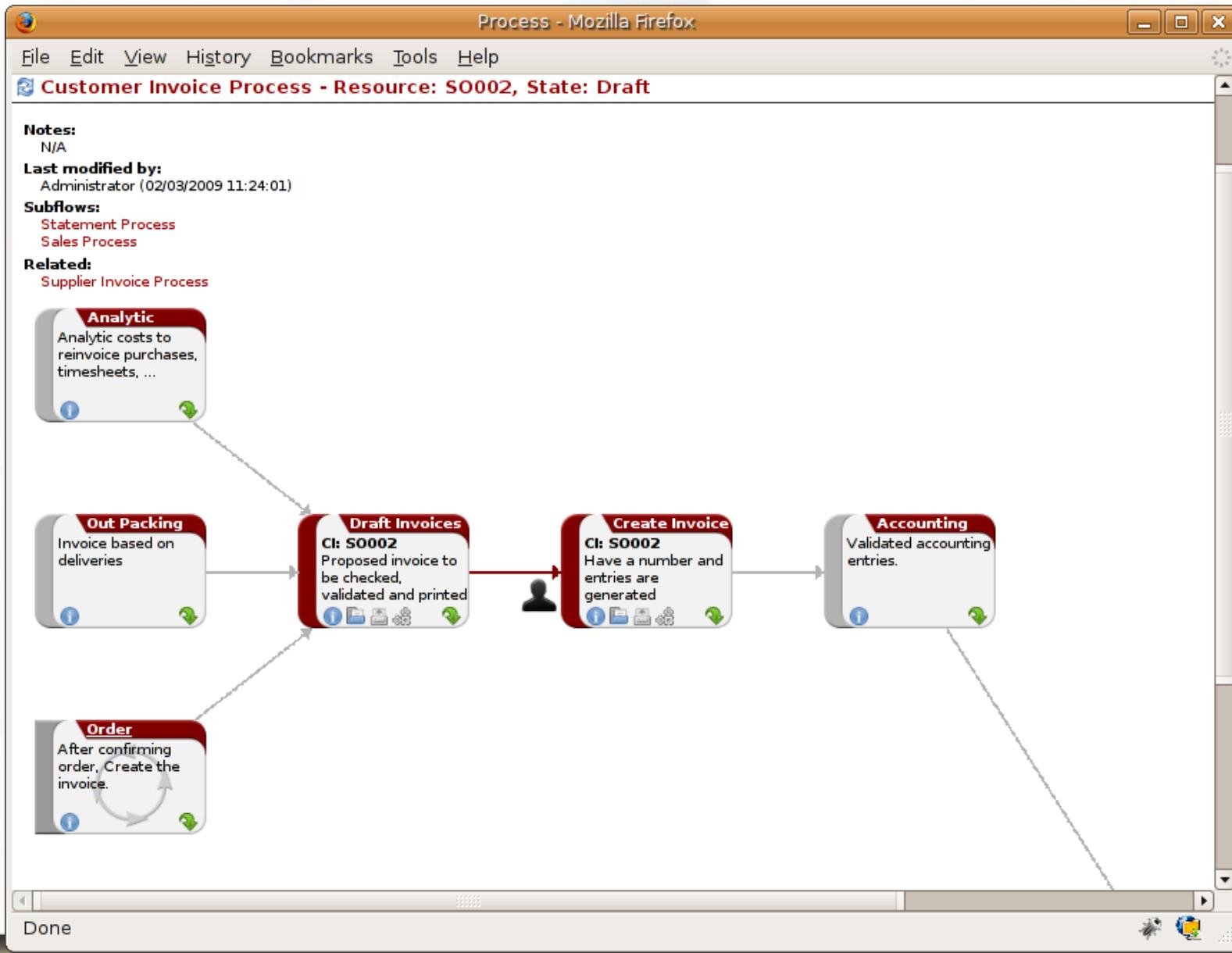
[CUSTOMIZE]

Copyright © 2007-TODAY Tiny ERP Pvt. Ltd. All Rights Reserved. More information on <http://openerp.com>.

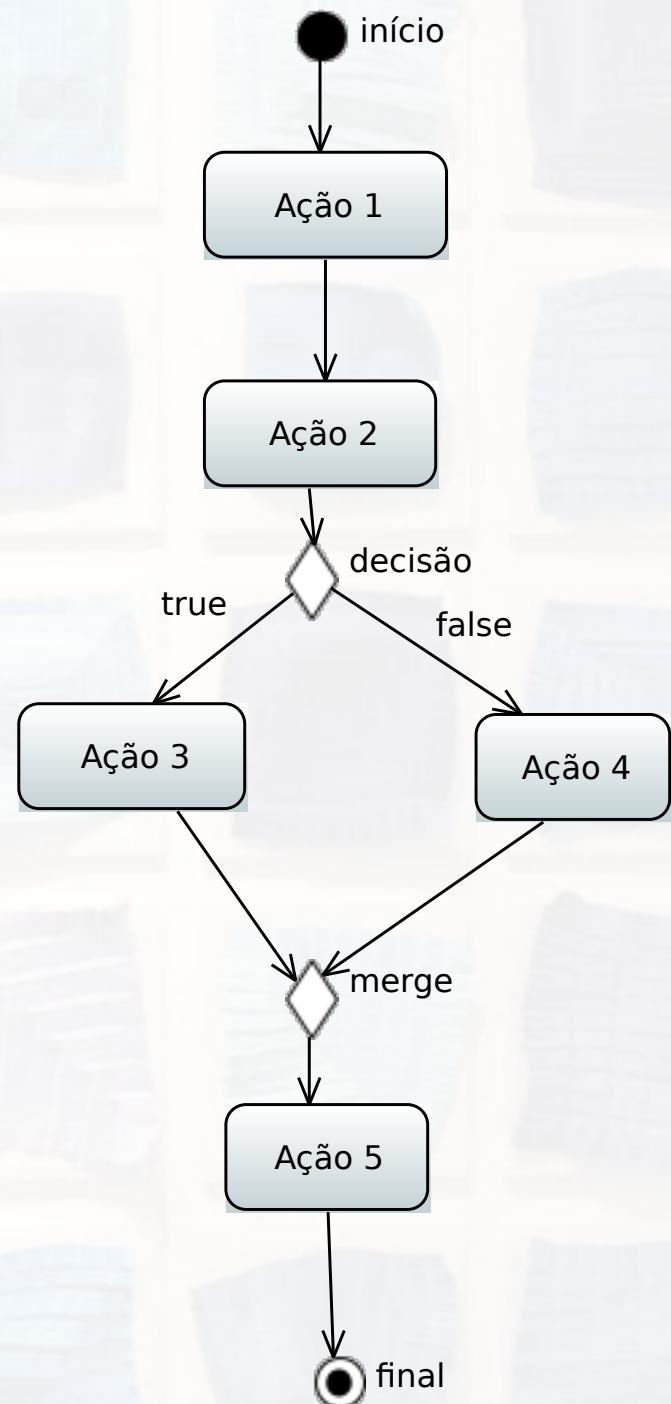
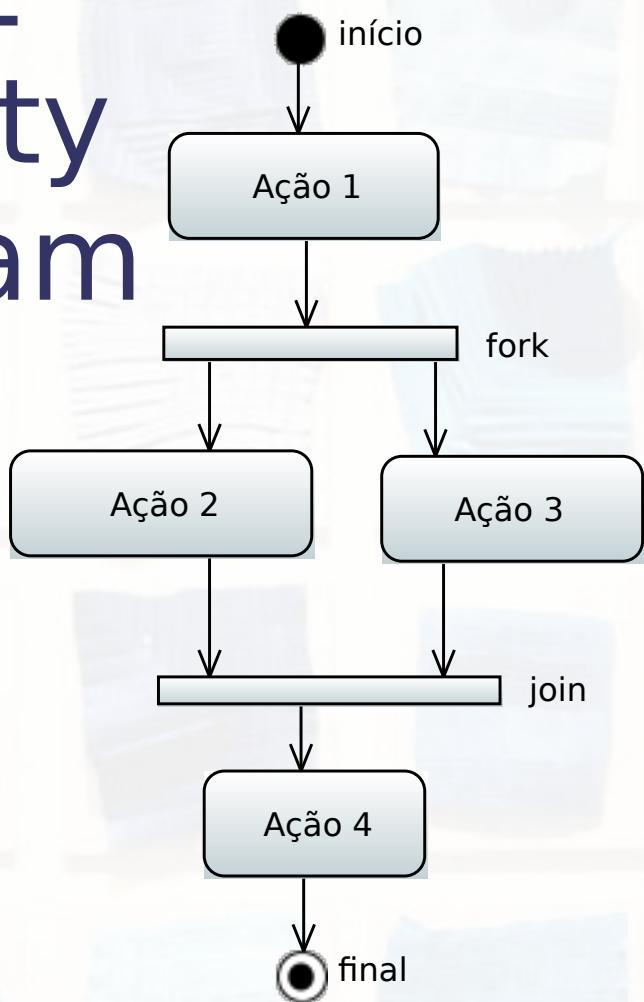
The web client is developed by Axelor (<http://axelor.com>) and Tiny (<http://tiny.be>)

Running Server: socket://localhost:8070 - database: screen

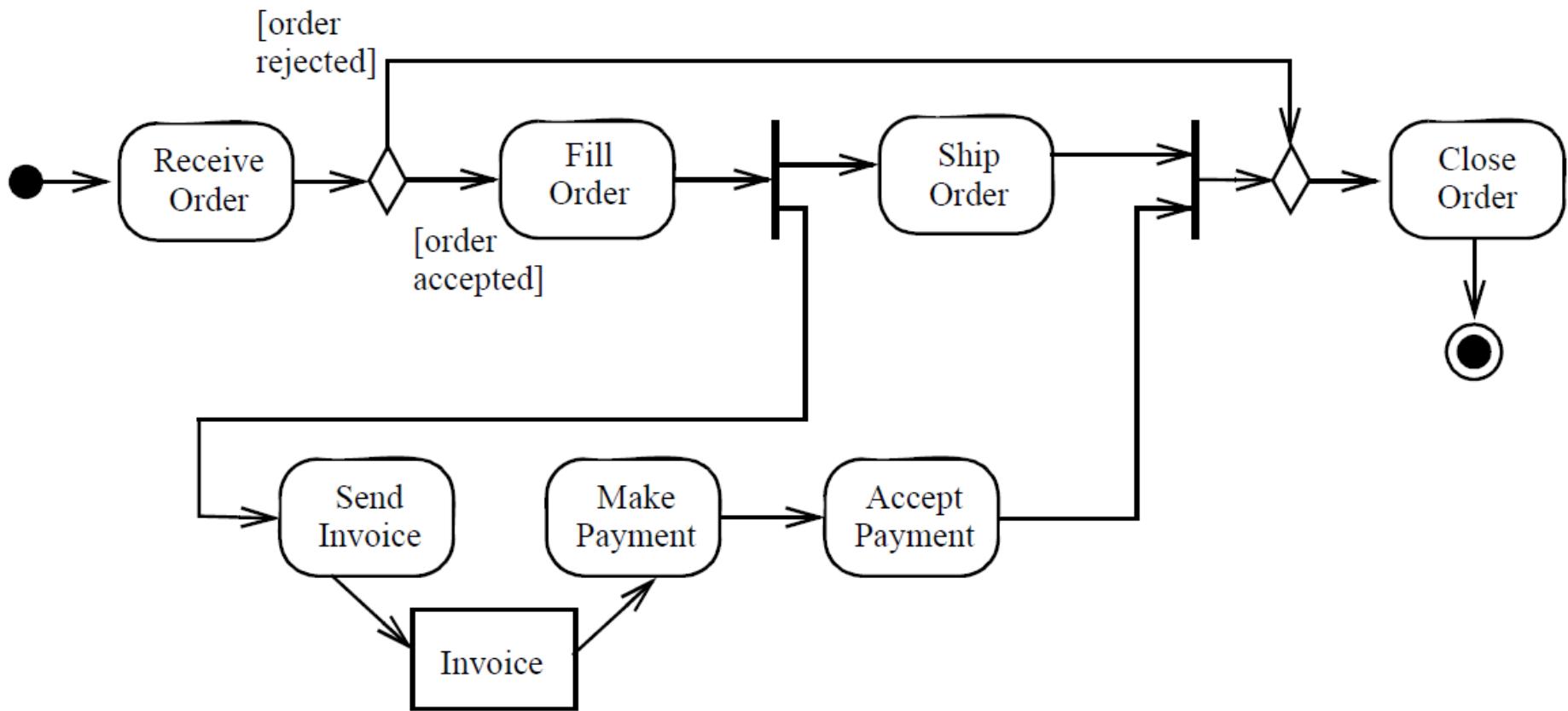
Open ERP Workflows



UML Activity Diagram

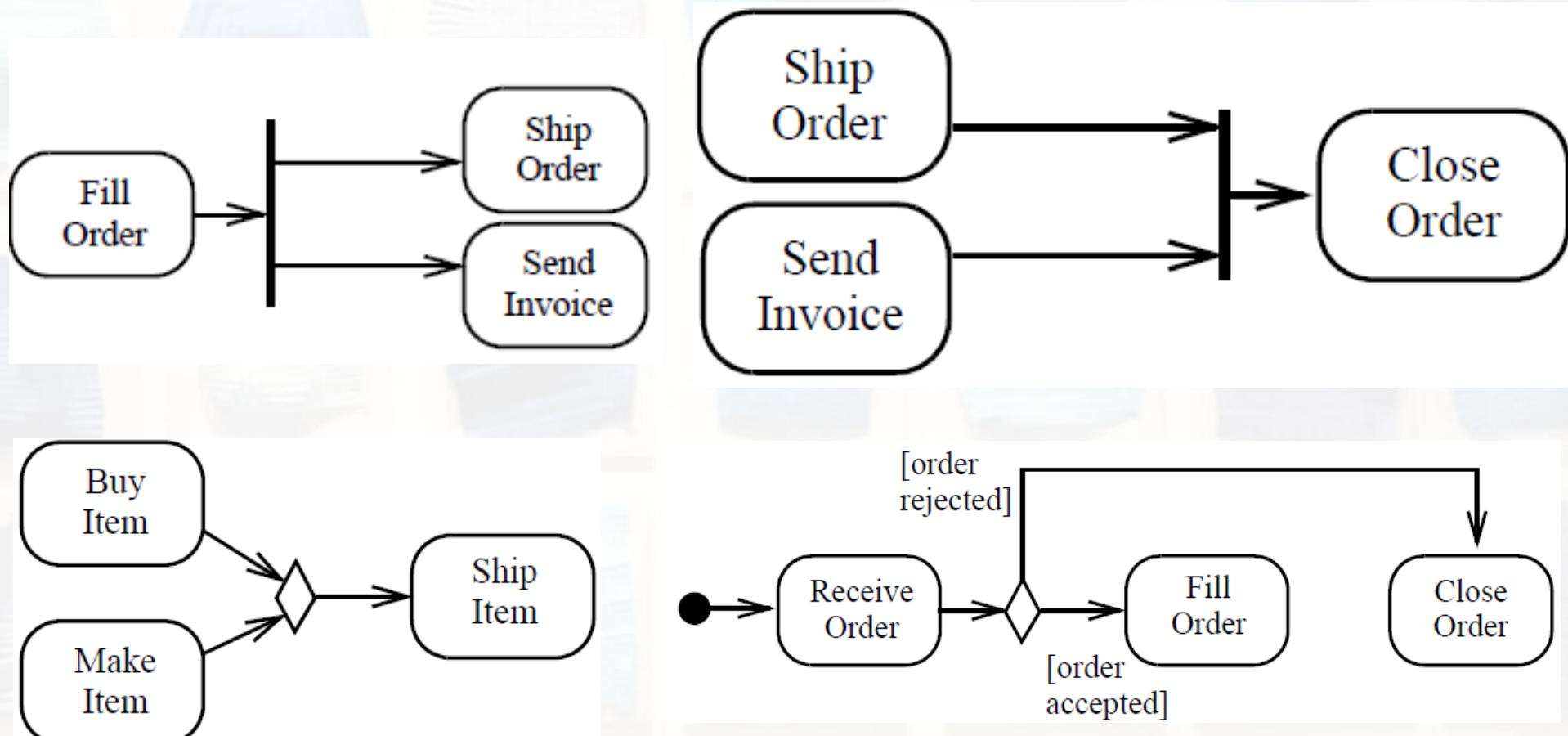


Exemplo de Compra



OMG Unified Modeling Language TM (OMG UML)
Version 2.5
<http://www.omg.org/spec/UML/2.5>

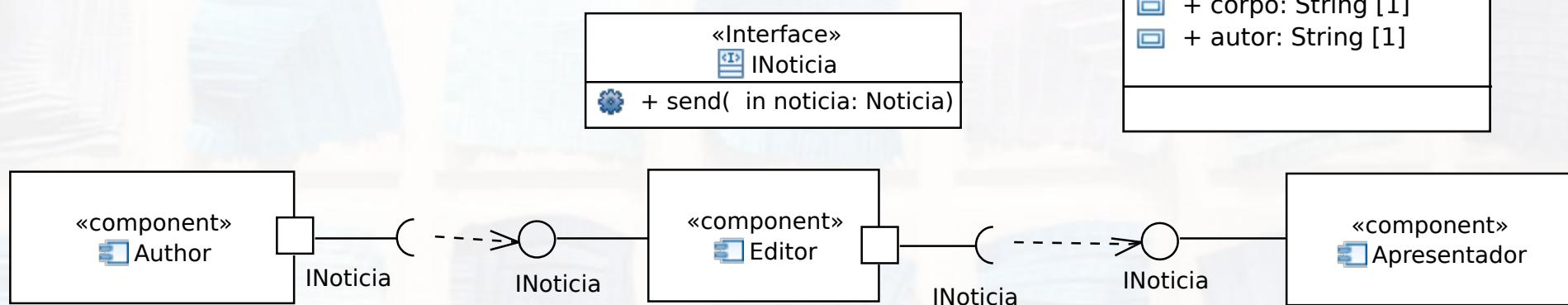
Exemplo de Compra



OMG Unified Modeling Language TM (OMG UML)
Version 2.5
<http://www.omg.org/spec/UML/2.5>

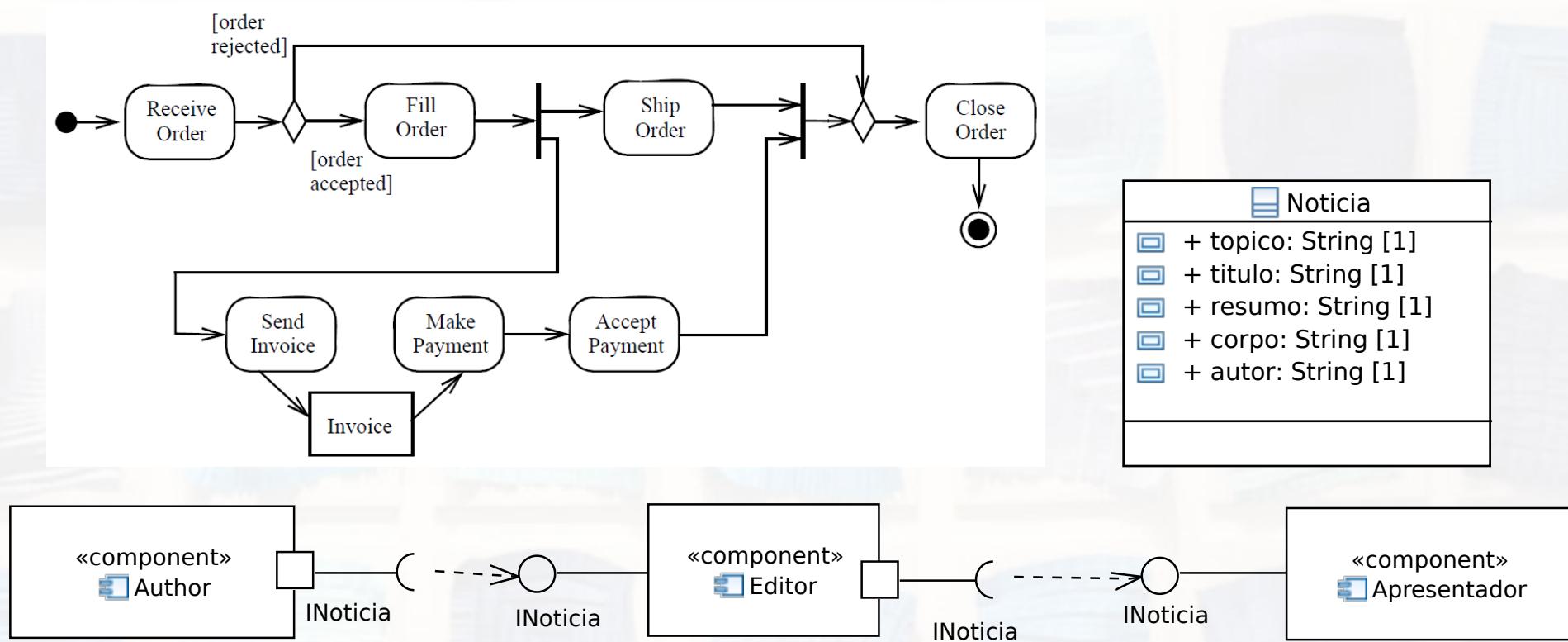
Tarefa 3

- Escreva o fluxo de atividades que interliga os componentes da Tarefa 2 desde a escrita da notícia, até sua publicação.
- Considere que pode-se enviar notícia pela Web ou por e-mail.

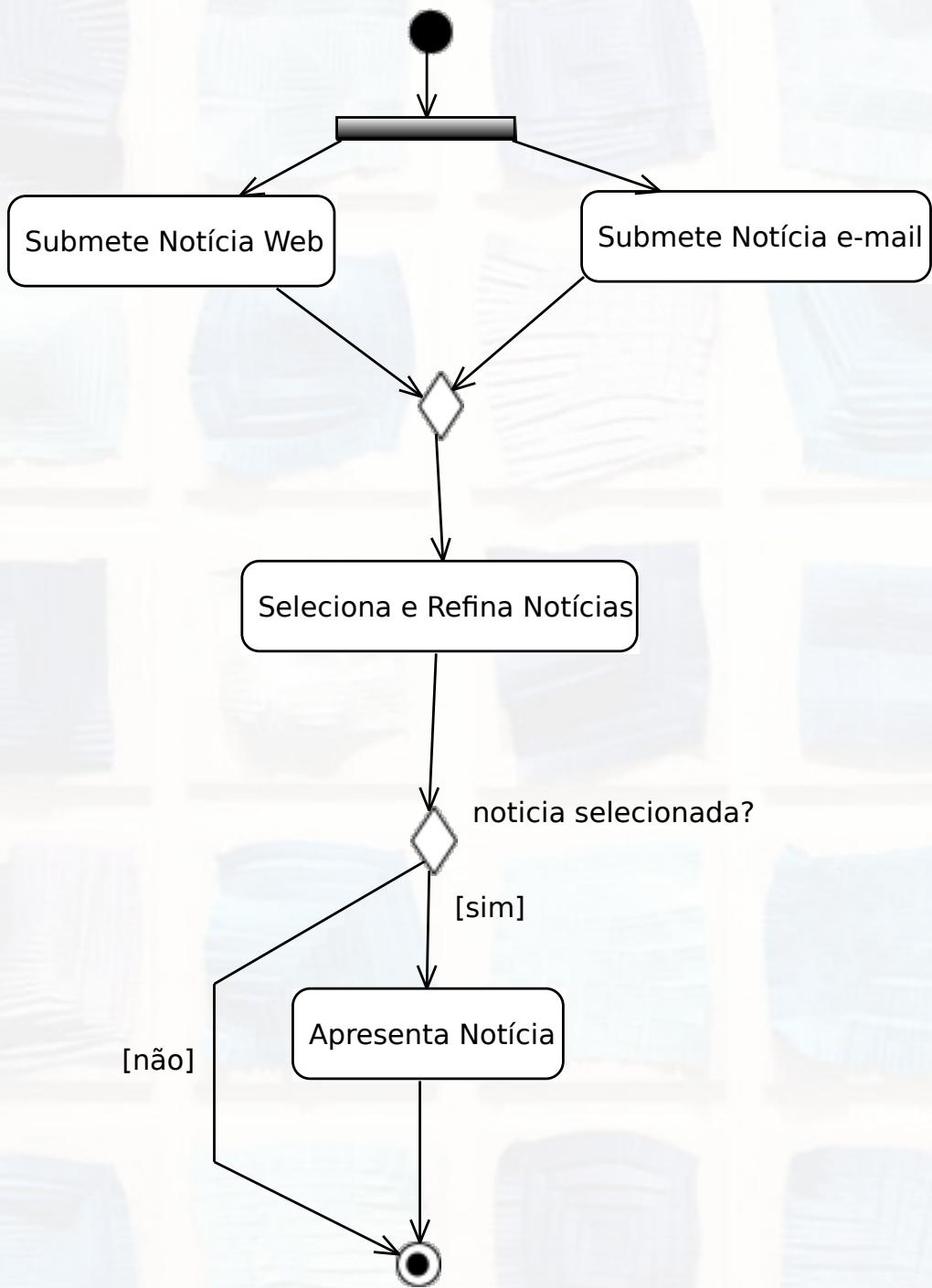


Tarefa 3

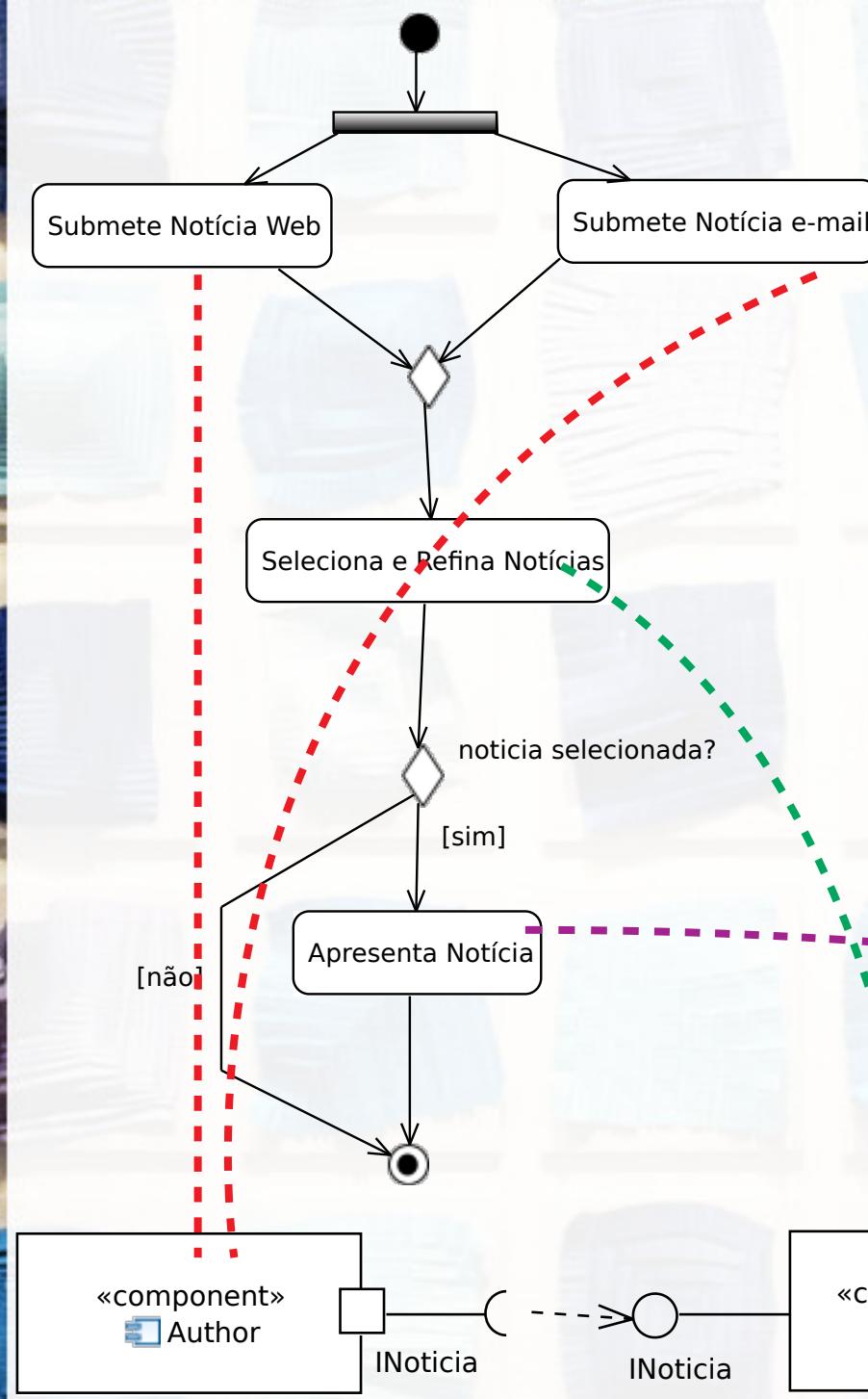
- Escreva o fluxo de atividades que interliga os componentes da Tarefa 2 desde a escrita da notícia, até sua publicação.
- Considere que pode-se enviar notícia pela Web ou por e-mail.



UML Activity Diagram



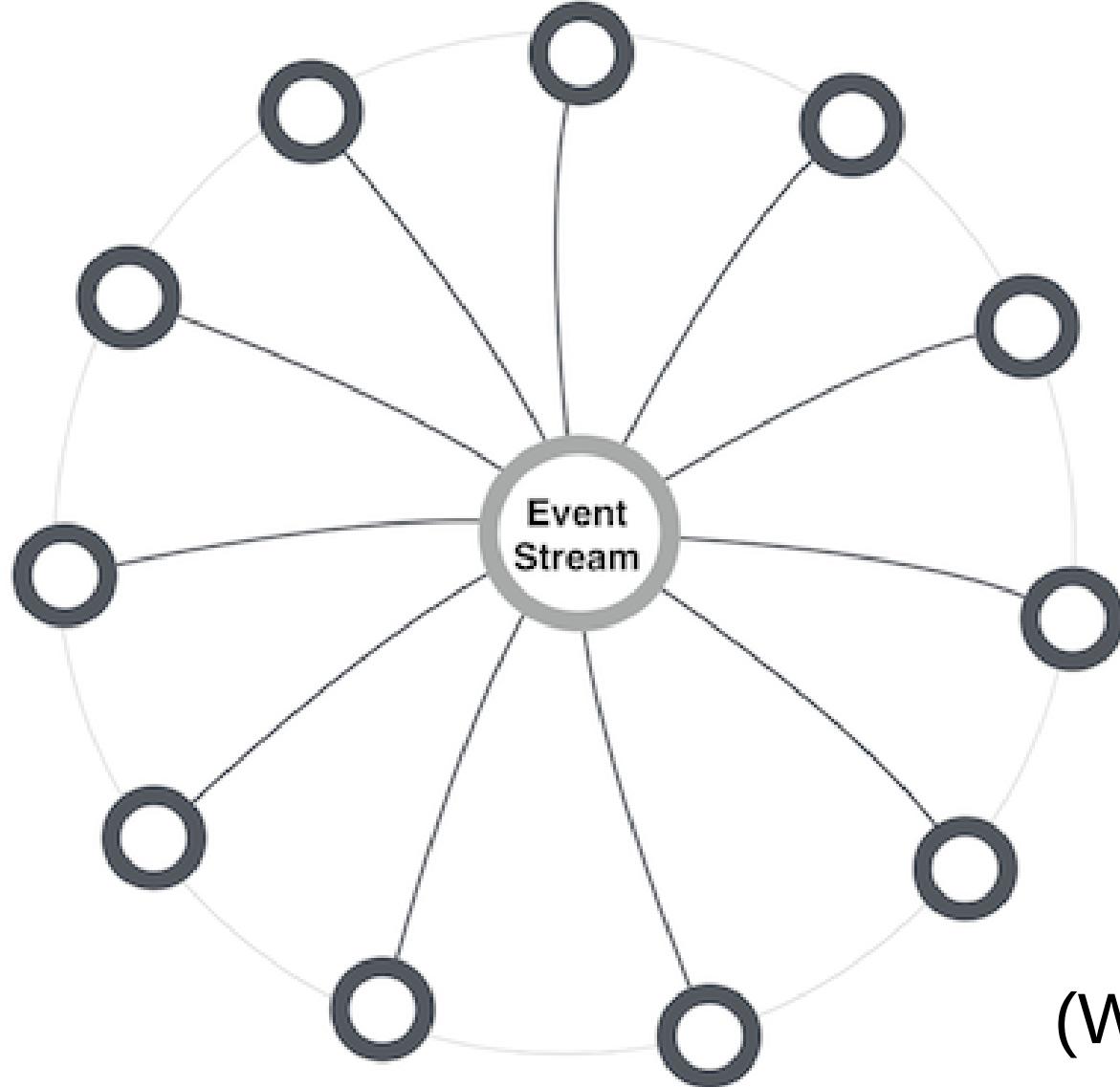
UML Activity Diagram



Coreografia

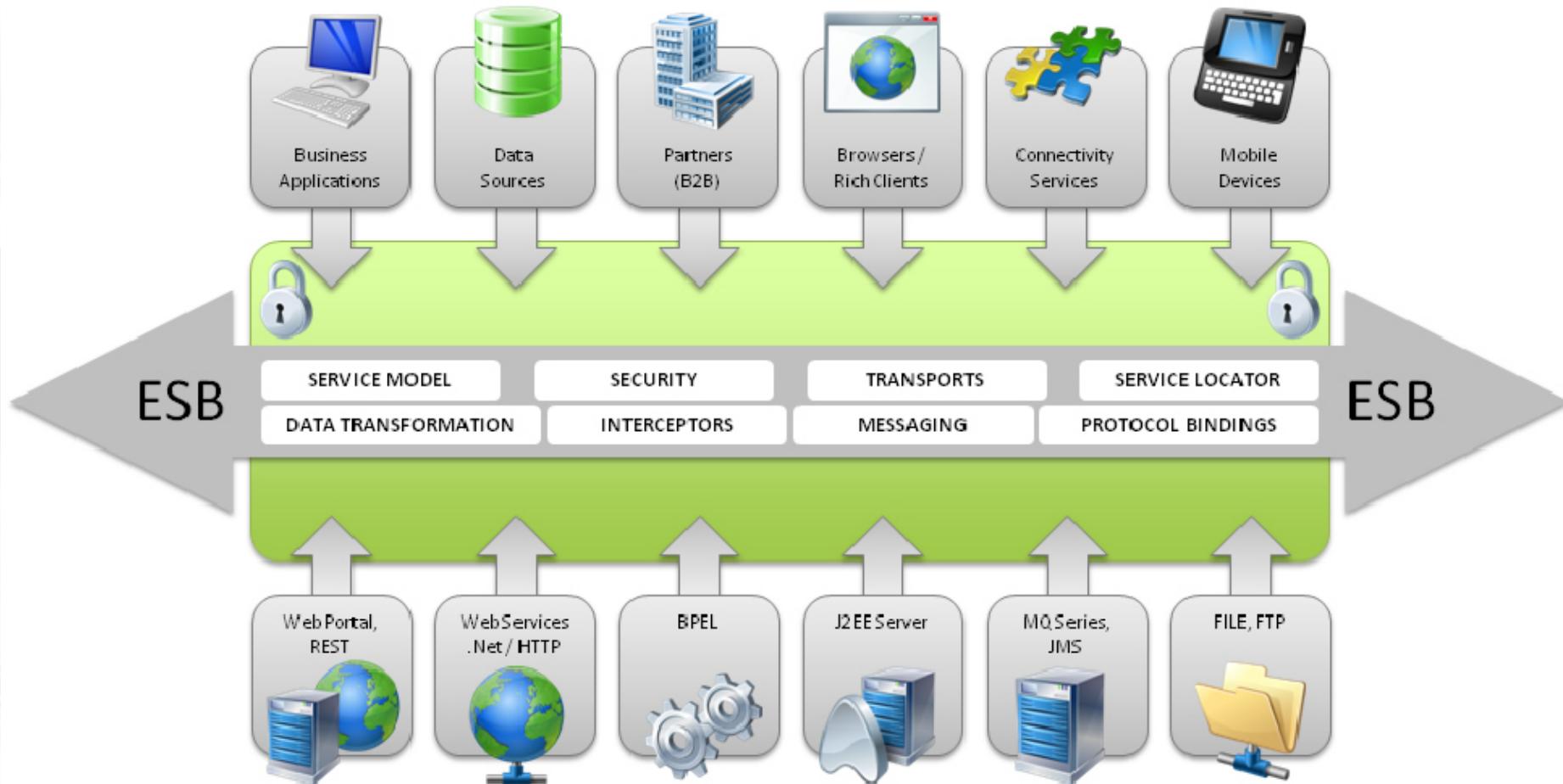


Coreografia



(Wolf, 2018)

Enterprise Service Bus (ESB)



Everything you need to know about Enterprise Service Bus (ESB)

Sanchit Agrawal | October 25, 2016

<https://www.hcltech.com/blogs/everything-you-need-know-about-enterprise-service-bus-esb>

Padrões de Mensagens

- ISO8583

- NACHA

- EDIFACT

- HL7

- SWIFT

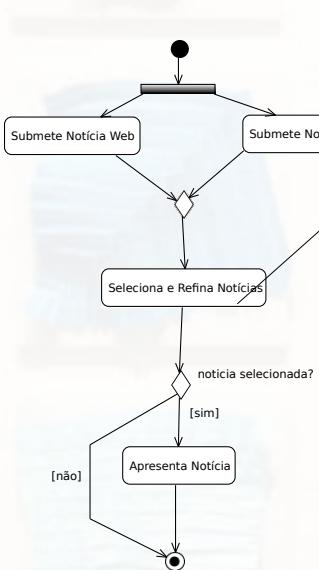
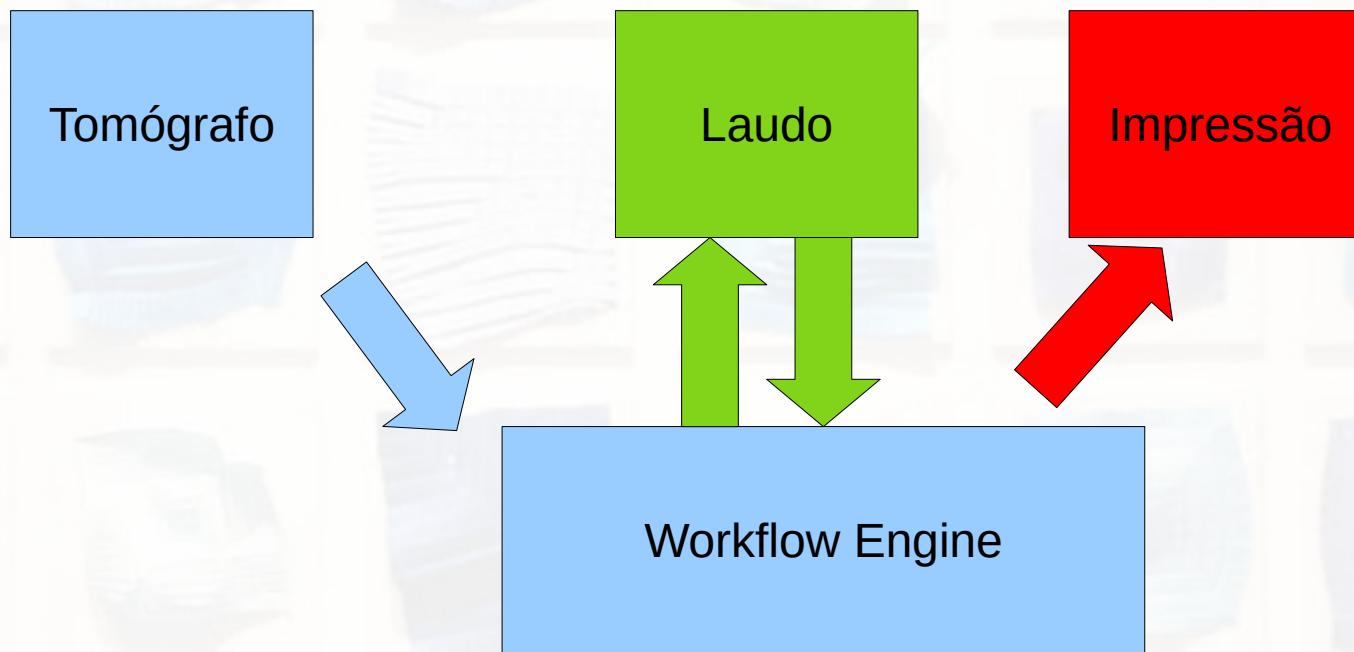
- TLOG

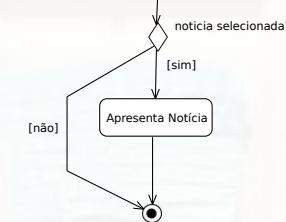
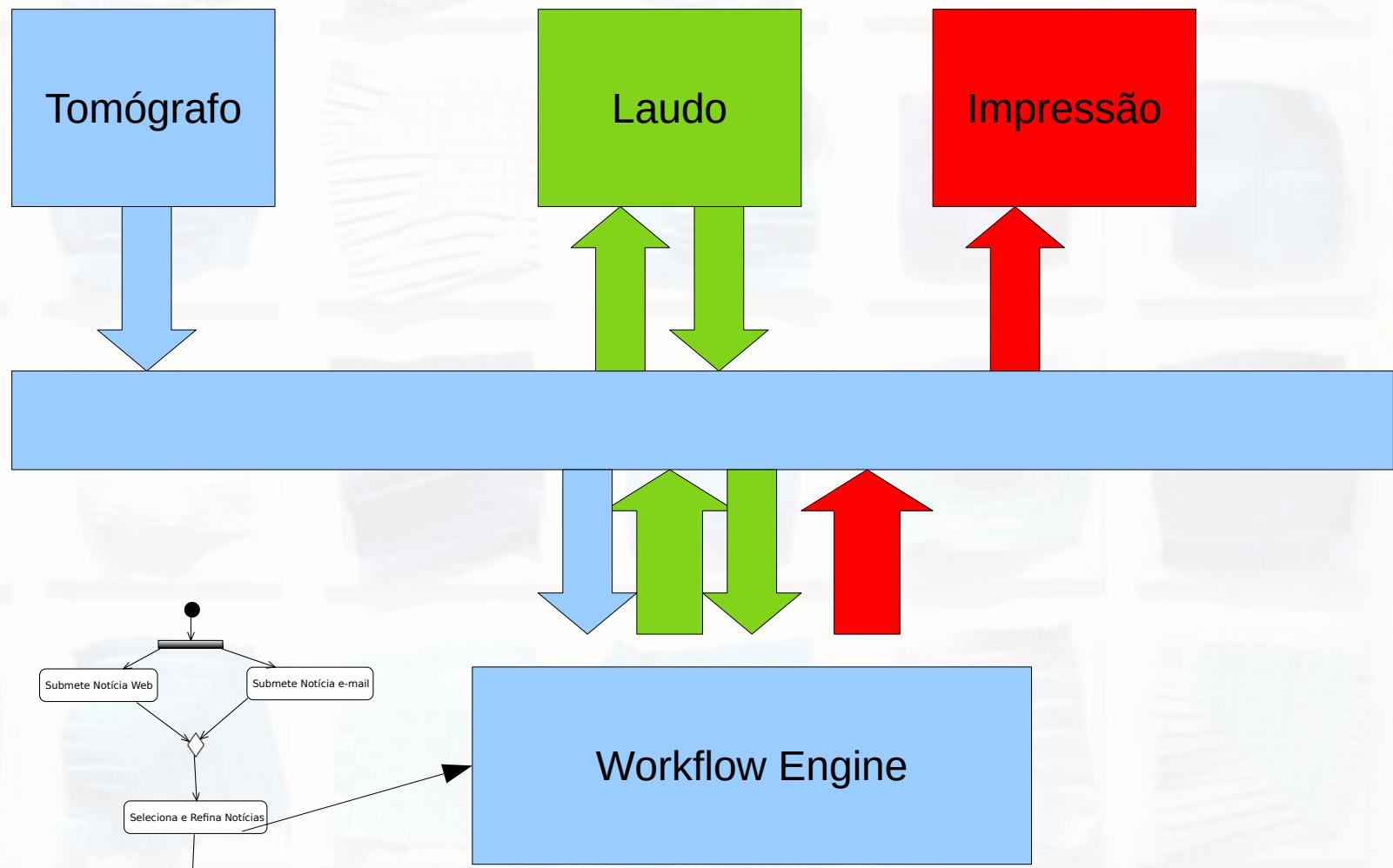
- X12

- ACORD

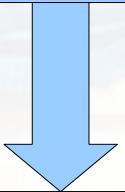
- FIX

- IDoc

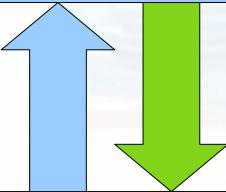




Tomógrafo

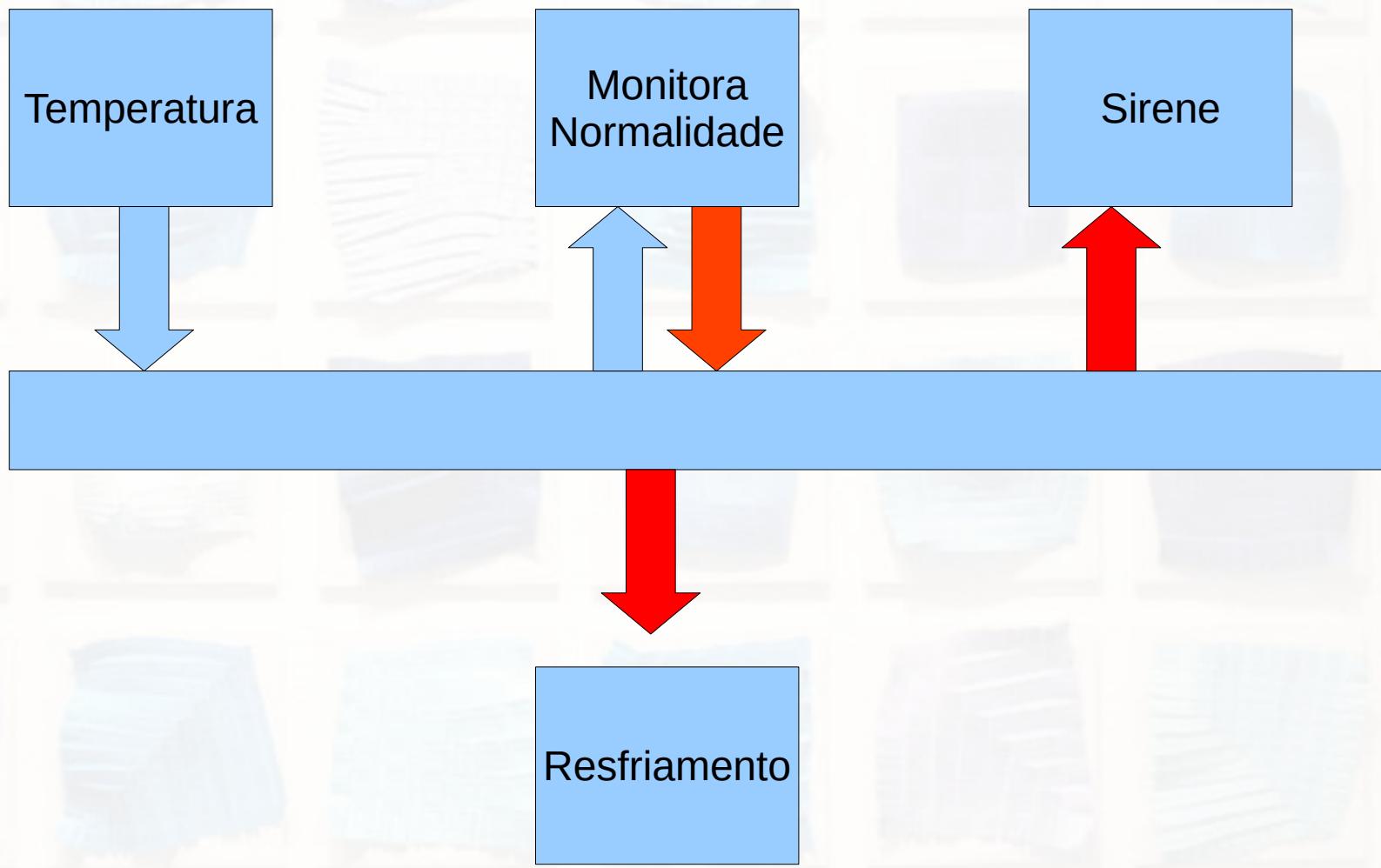


Laudo

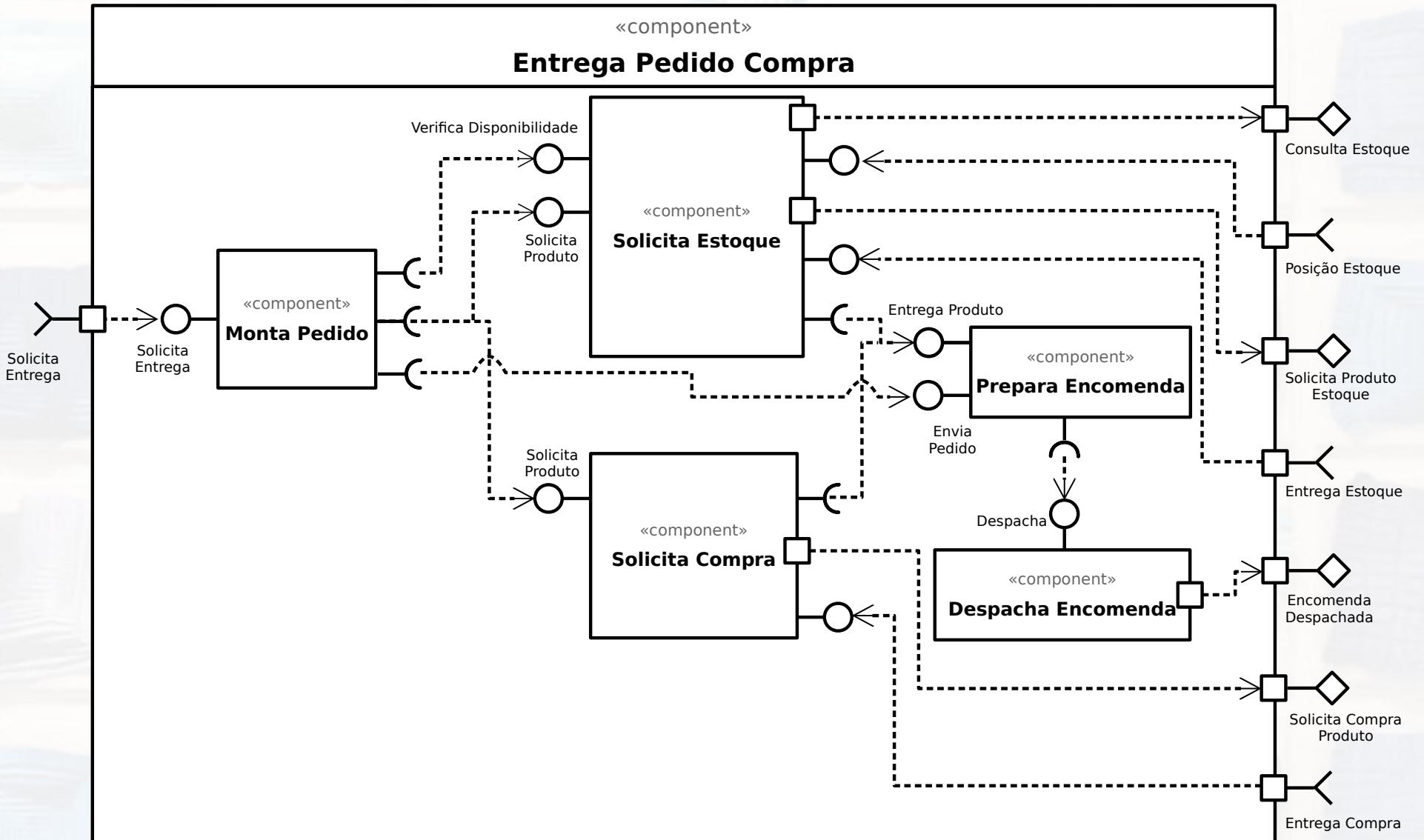


Impressão





Compondo em Dois Níveis



Monitorando Sensores



Estilos Arquiteturais Decomposição Modular Pipe & Filter

Pipe & Filter

- Bastante popular em sistemas operacionais UNIX-like
- Processo incremental
 - vai gerando os dados de saída, sem esperar que a entrada de dados se complete (Garlan, 1993).
- Invariantes (Garlan, 1993)
 - entidades independentes
 - identidades de entrada e saída desconhecidas
 - especificação local

Pipe & Filter

■ *Filter* (componente)

- Lê fluxos de dados de entrada e produz seus resultados como fluxos de dados de saída.



■ *Pipe* (conector)

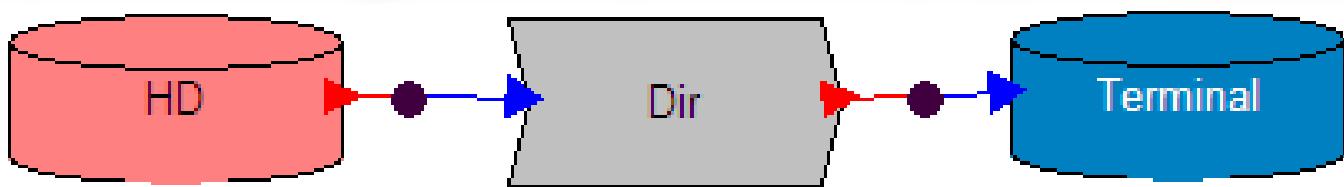
- Conduzem o fluxo, conectando o fluxo de saída de um filter ao fluxo de entrada de outro filter.



Pipe & Filter UNIX-like

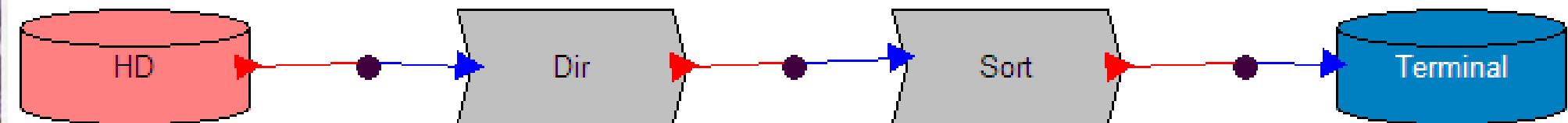
■ Lista nome dos arquivos

□ dir /b



Pipe & Filter UNIX-like

- Operador de pipe no DOS e Unix: |
- Lista nome dos arquivos “pipe” coloca em ordem alfabética
 - dir /b | sort



Pipe & Filter Unix-like

■ Lista nome dos arquivos “pipe” coloca em ordem alfabética “pipe” recorta aqueles que têm o trecho “Win”

□ `dir /b | sort | grep "Win"`

■ Redireciona saída (pipe) no DOS: `>`

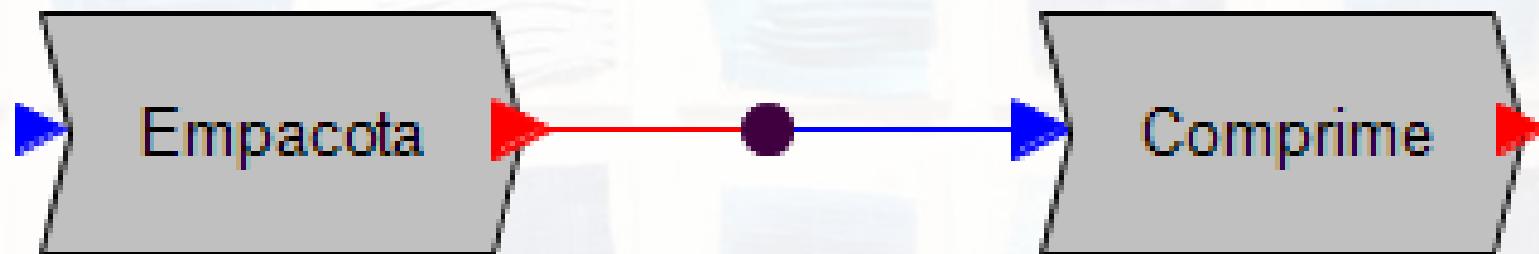
■ Mesmo anterior com saída para arquivo “resultado.txt”

□ `dir /b | sort | grep "Win" >resultado.txt`

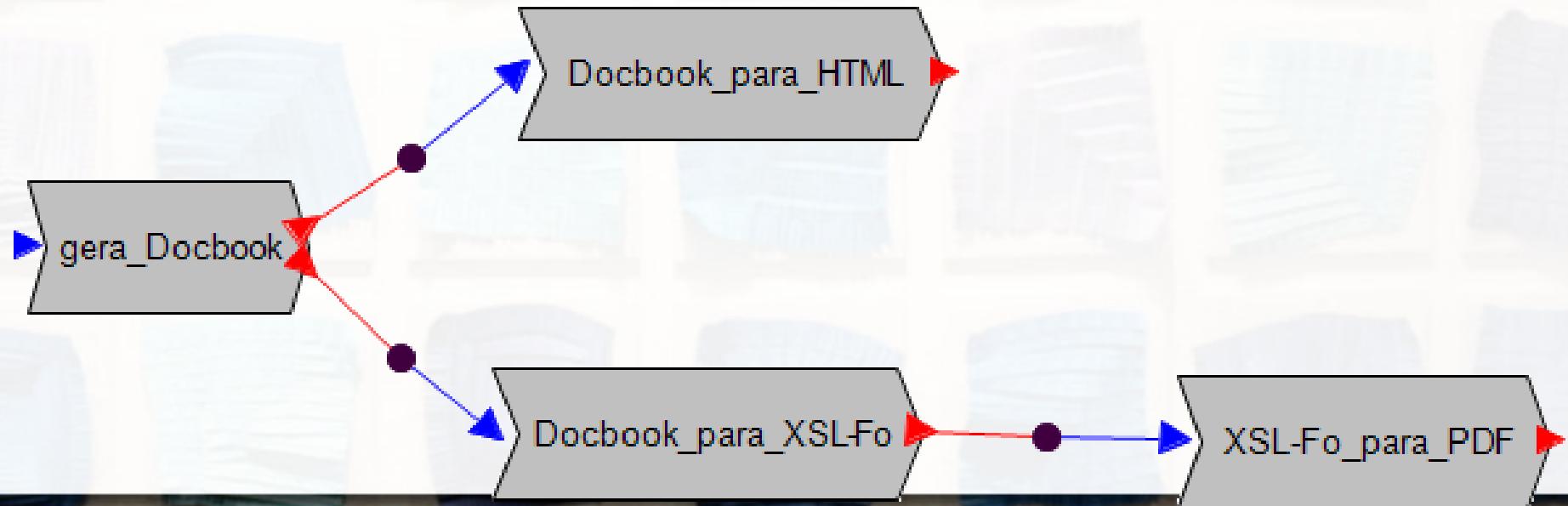


Pipe & Filter Exemplos

■ Empacotando e comprimindo



■ Docbook



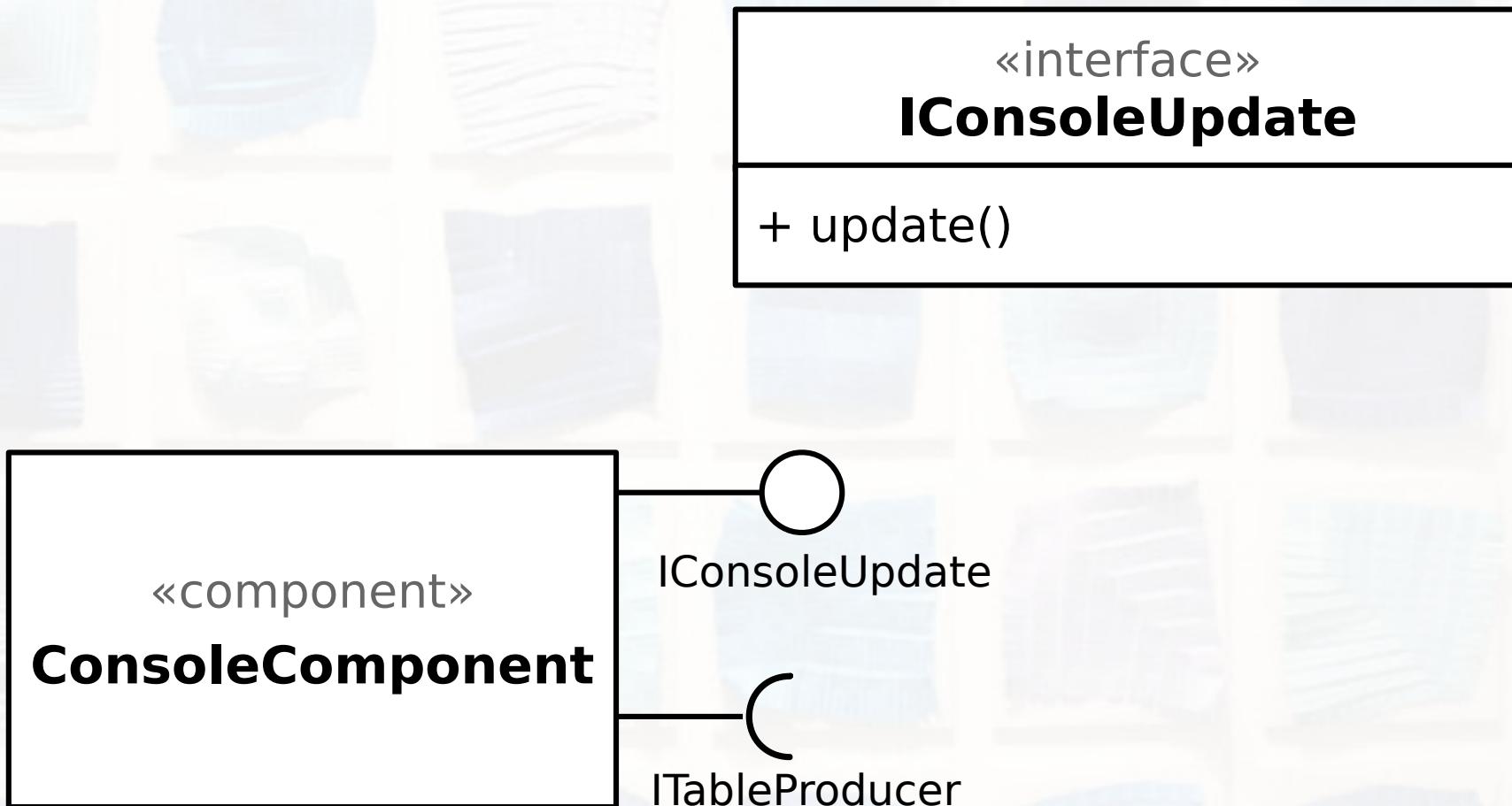
Vantagens do Pipe & Filter

- Suporta reúso de transformações
- Organização intuitiva para a comunicação das partes
- Fácil de adicionar novas transformações
- Relativamente simples de implementar tanto em sistemas concorrentes como sequenciais.”¹

(Sommerville, 2007)

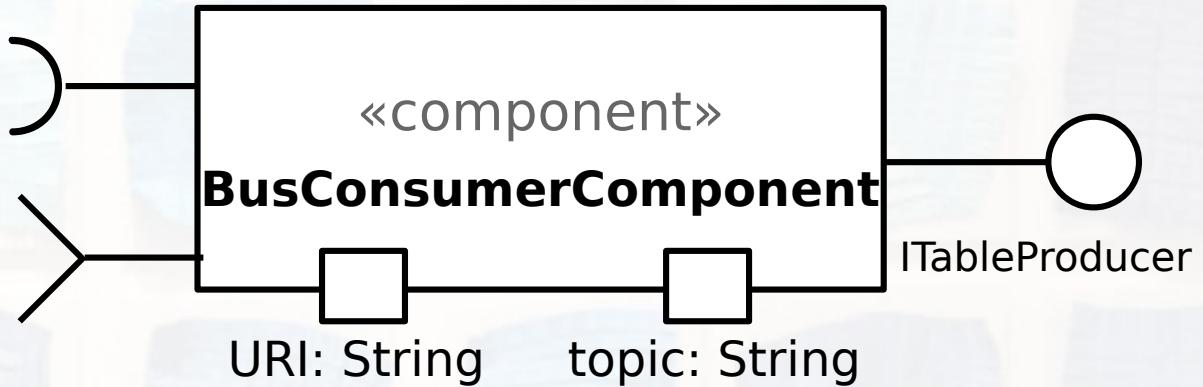
1. “- Supports transformation reuse.
- Intuitive organisation for stakeholder communication.
- Easy to add new transformations.
- Relatively simple to implement as either a concurrent or sequential system.” (Sommerville,

Componente Console



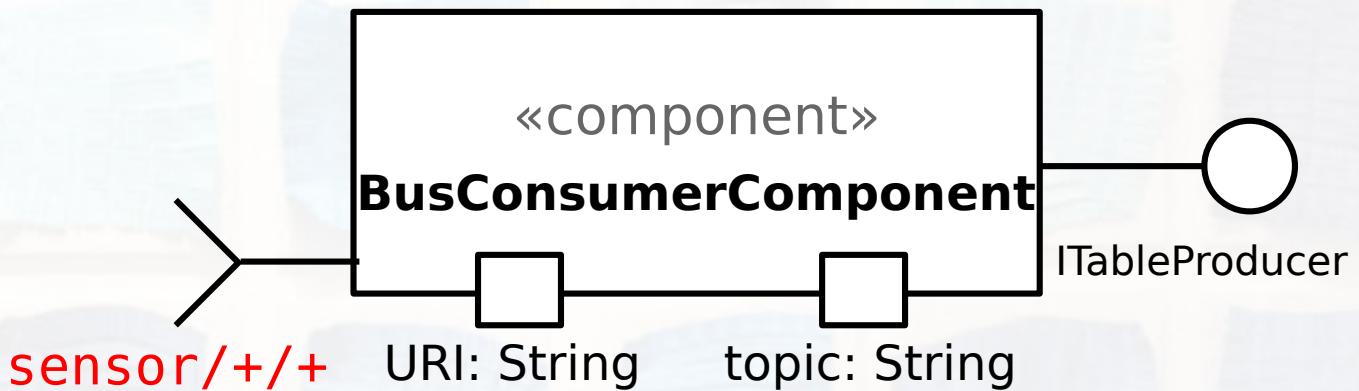
Conectando Componentes

```
IBusConsumer bc = new BusConsumerComponent();
```



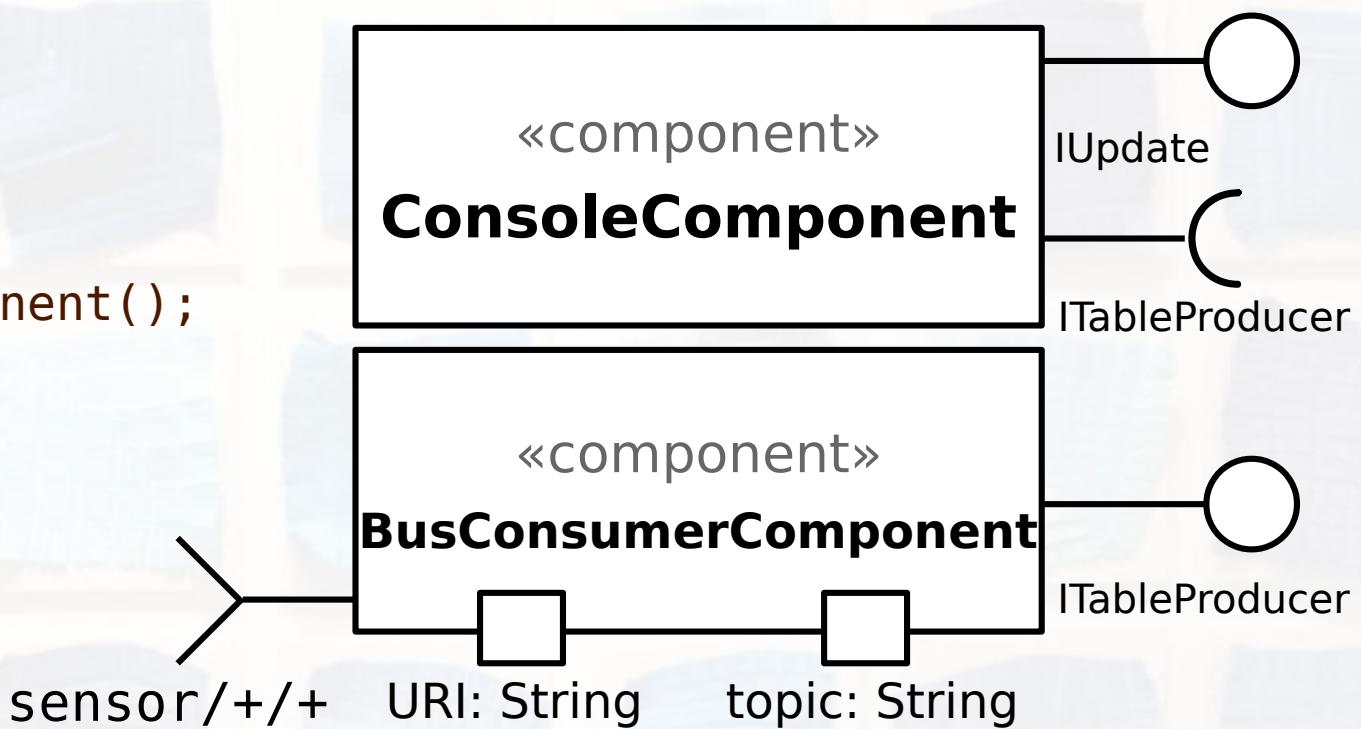
Conectando Componentes

```
IBusConsumer bc = new BusConsumerComponent();  
bc.setBusURI("tcp://localhost:1883");  
bc.setTopic("sensor/+/-");  
bc.setBlockSize(10);  
bc.setVerbose(2);
```



Conectando Componentes

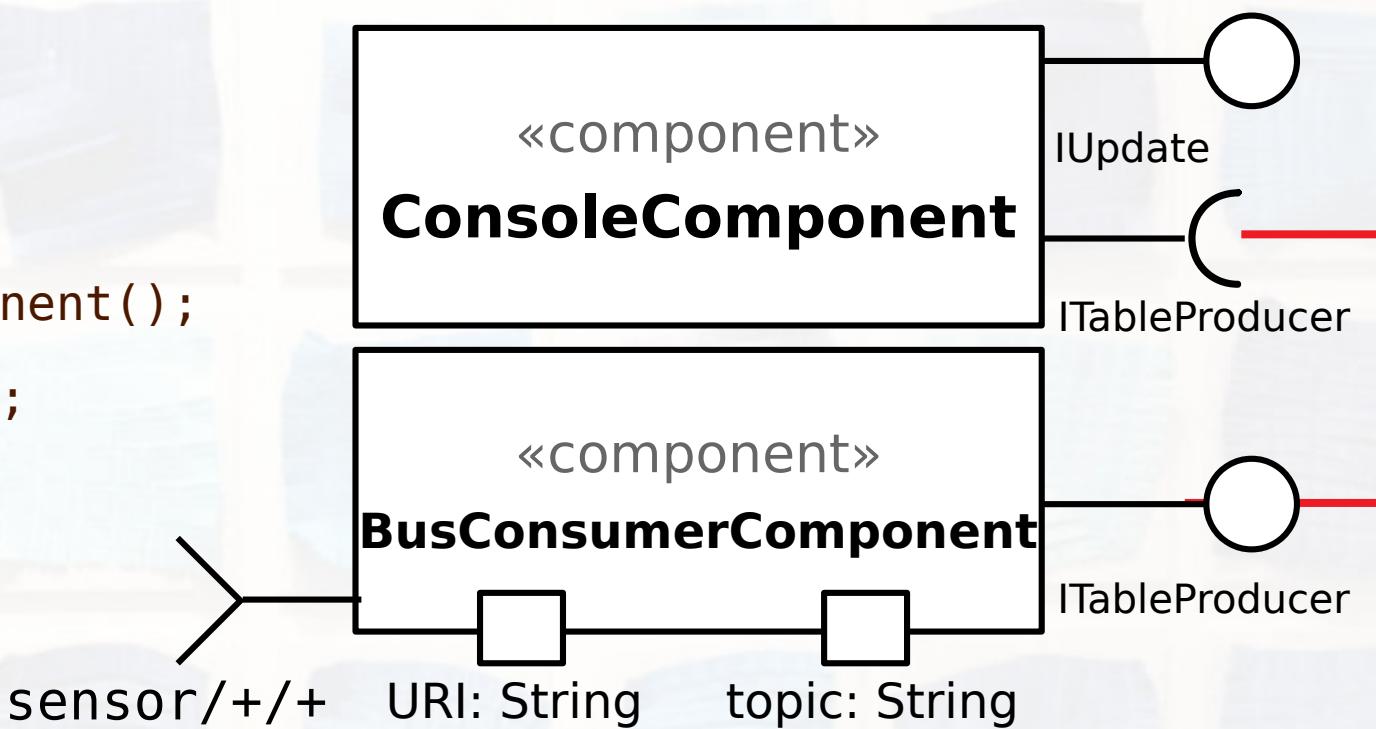
```
IBusConsumer bc = new BusConsumerComponent();  
  
bc.setBusURI("tcp://localhost:1883");  
  
bc.setTopic("sensor/+/+");  
  
bc.setBlockSize(10);  
  
bc.setVerbose(2);  
  
IConsole console =  
    new ConsoleComponent();
```



Conectando Componentes

```
IBusConsumer bc = new BusConsumerComponent();  
bc.setBusURI("tcp://localhost:1883");  
bc.setTopic("sensor/+/+");  
bc.setBlockSize(10);  
bc.setVerbose(2);
```

```
IConsole console =  
    new ConsoleComponent();  
console.connect(bc);
```



Conectando Componentes

```
IBusConsumer bc = new BusConsumerComponent();
```

```
bc.setBusURI("tcp://localhost:1883");
```

```
bc.setTopic("sensor/+/+");
```

```
bc.setBlockSize(10);
```

```
bc.setVerbose(2);
```

```
IConsole console =  
    new ConsoleComponent();
```

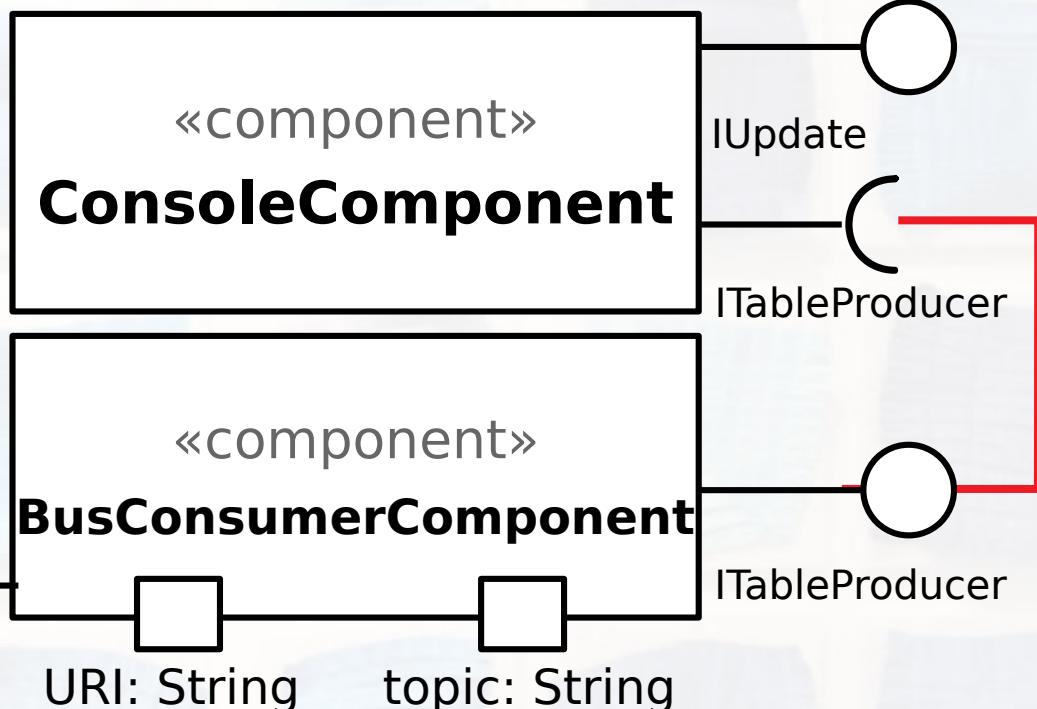
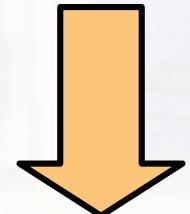
```
console.connect(bc);
```

```
bc.connect(console);
```

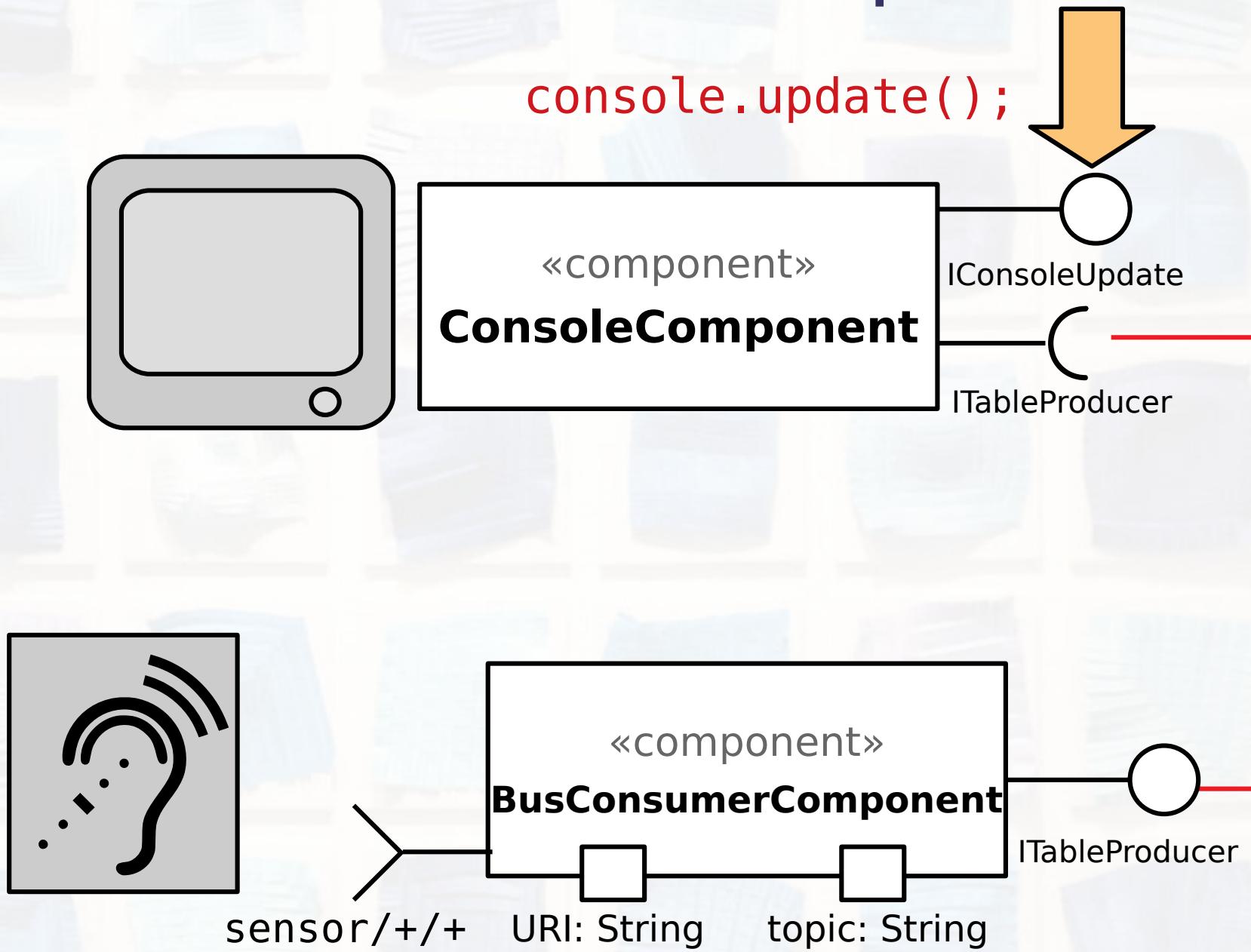
sensor/+/+

URI: String

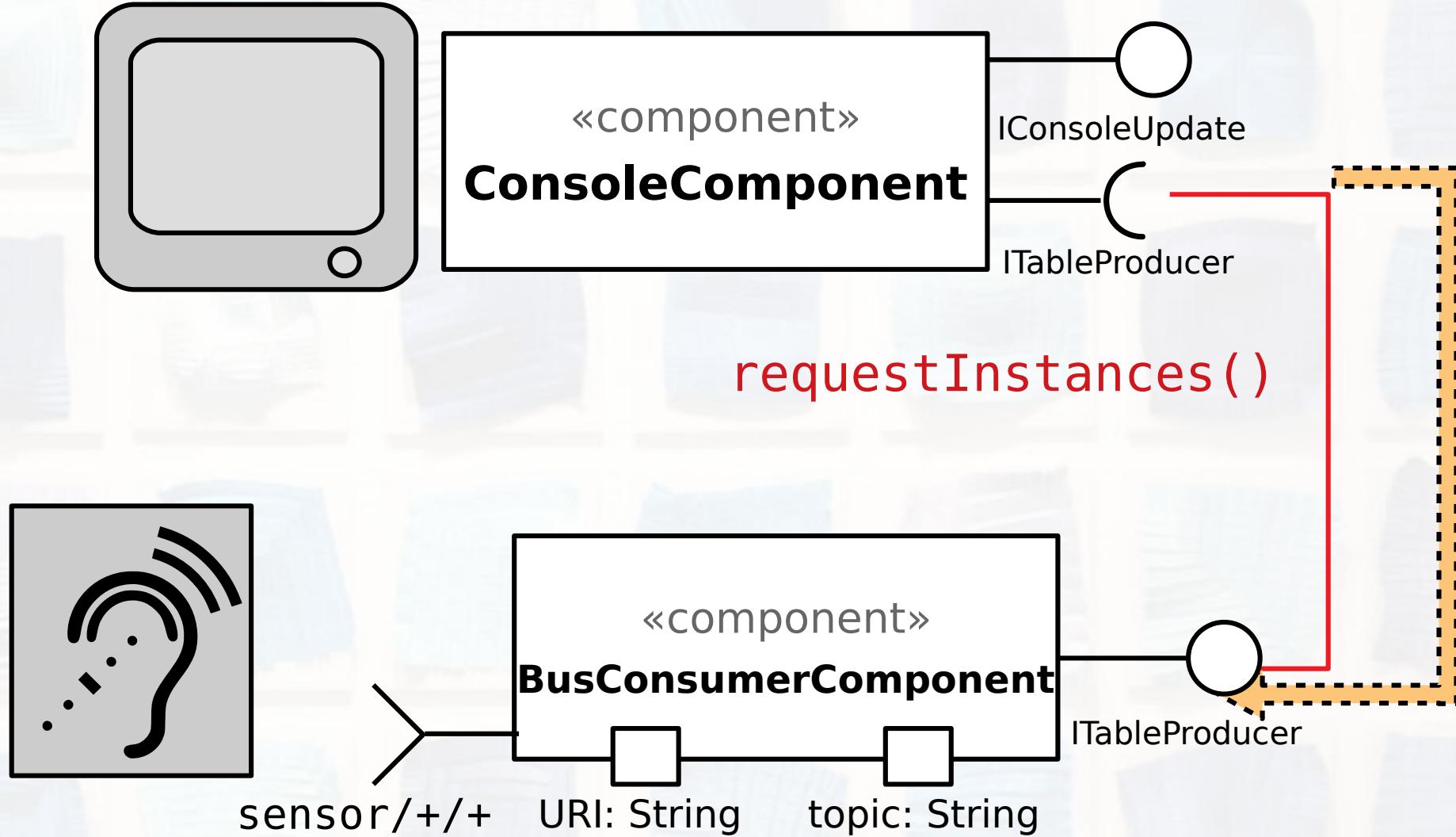
topic: String



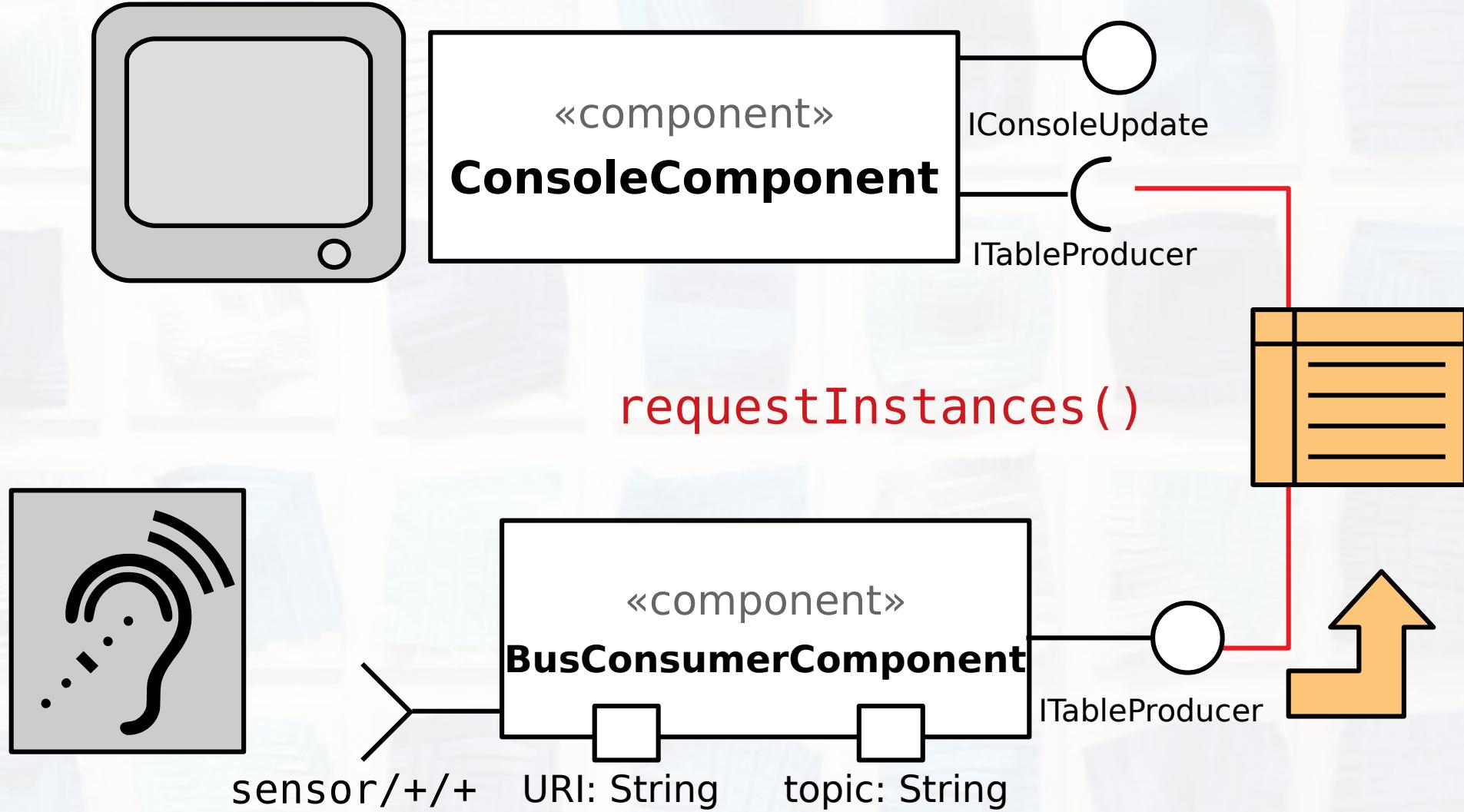
Conectando Componentes



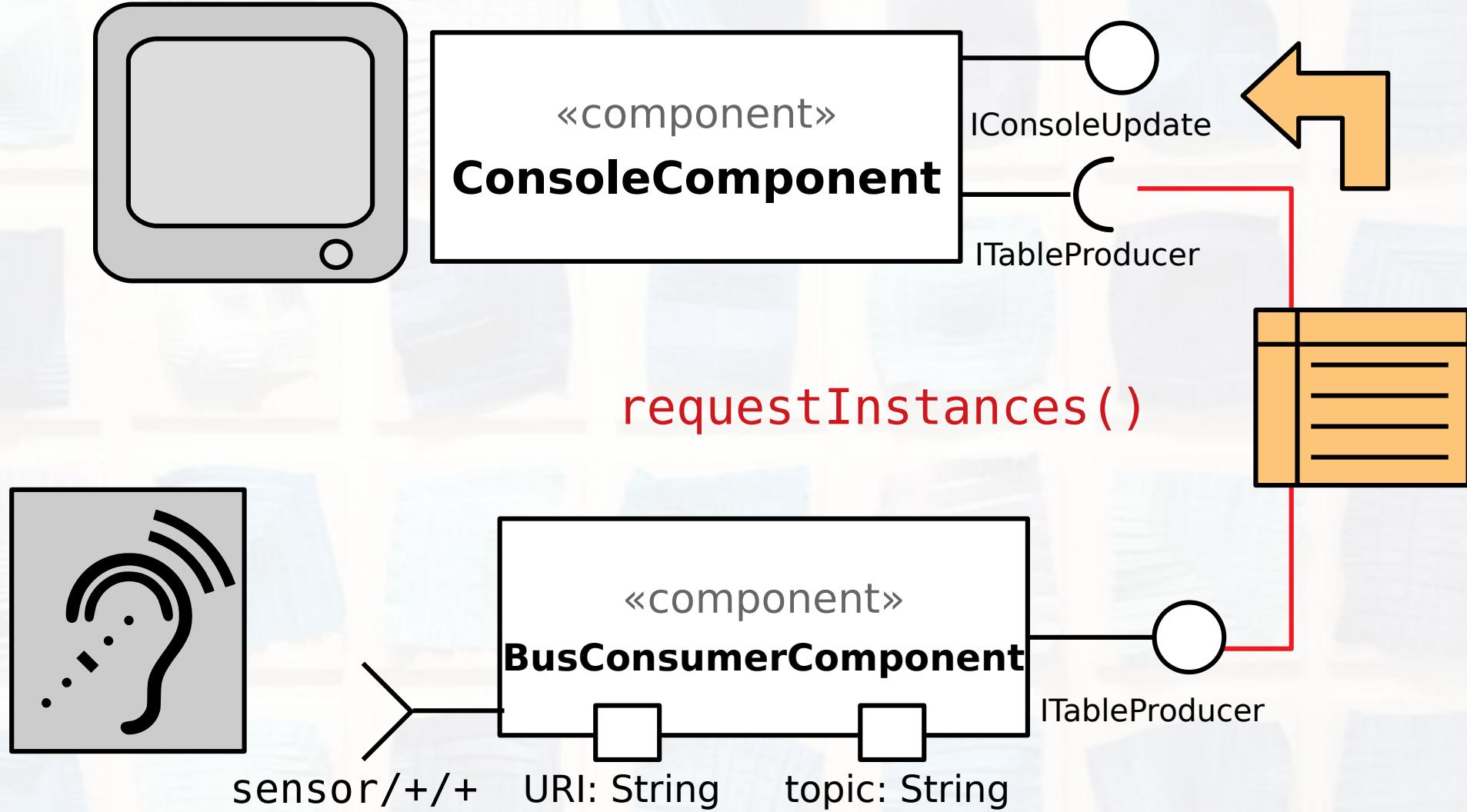
Conectando Componentes



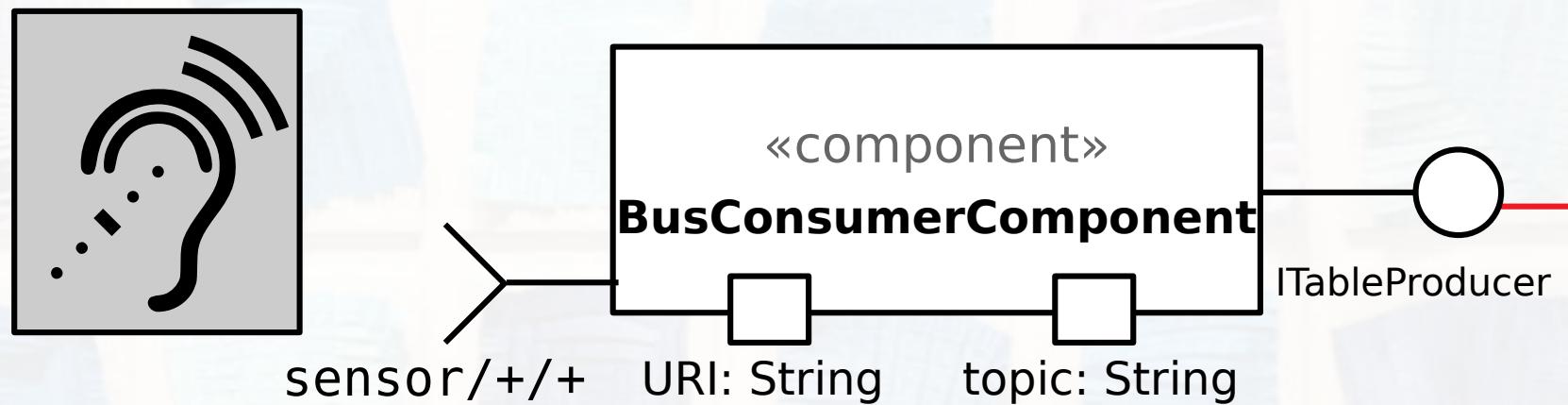
Conectando Componentes



Conectando Componentes

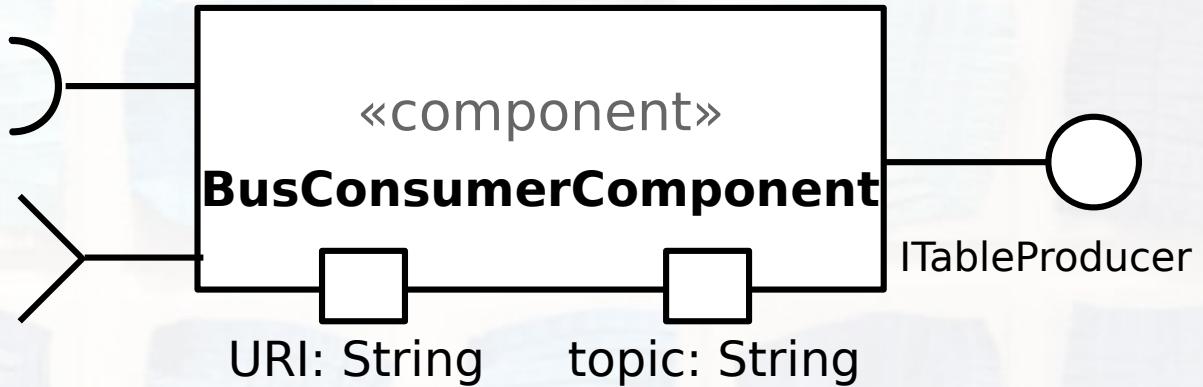


Conectando Componentes



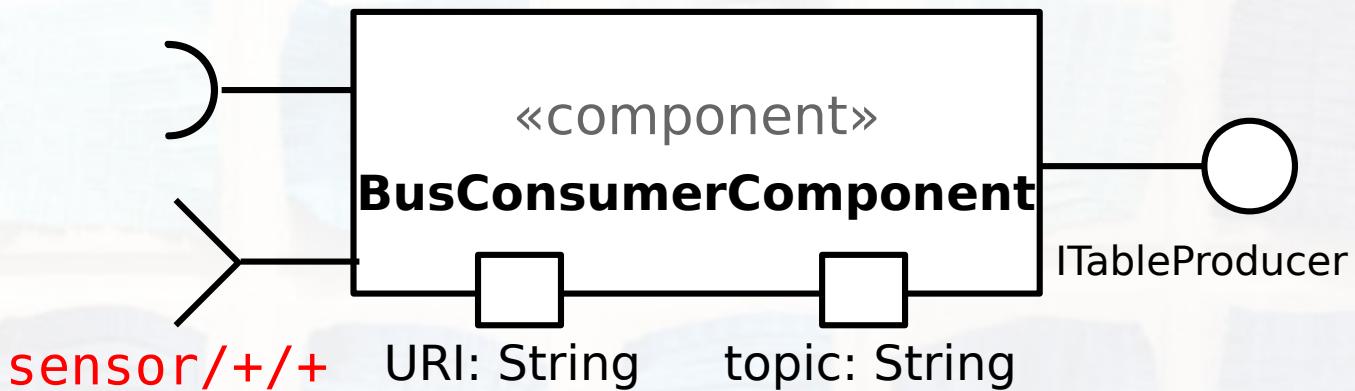
Conectando Componentes

```
IBusConsumer bc = new BusConsumerComponent();
```



Conectando Componentes

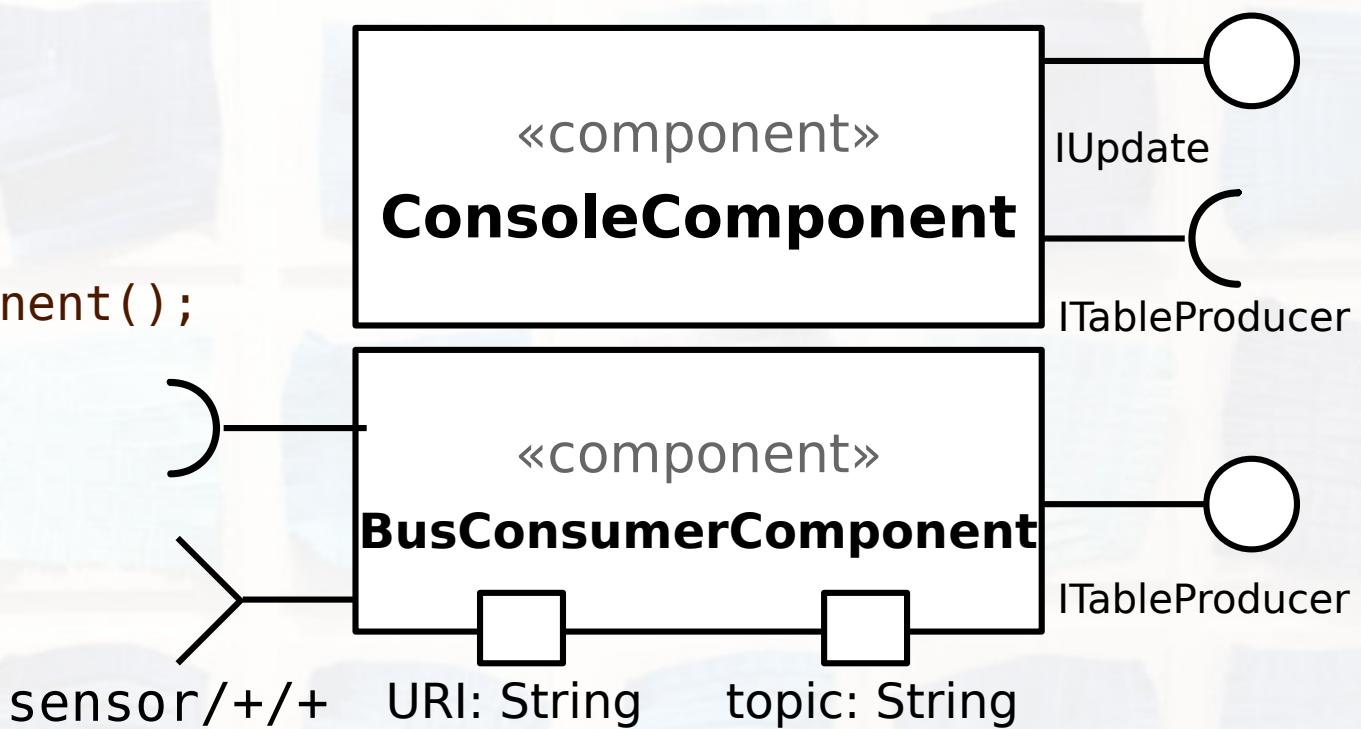
```
IBusConsumer bc = new BusConsumerComponent();  
bc.setBusURI("tcp://localhost:1883");  
bc.setTopic("sensor/+/-");  
bc.setBlockSize(10);  
bc.setVerbose(2);
```



Conectando Componentes

```
IBusConsumer bc = new BusConsumerComponent();  
  
bc.setBusURI("tcp://localhost:1883");  
  
bc.setTopic("sensor/+/+");  
  
bc.setBlockSize(10);  
  
bc.setVerbose(2);
```

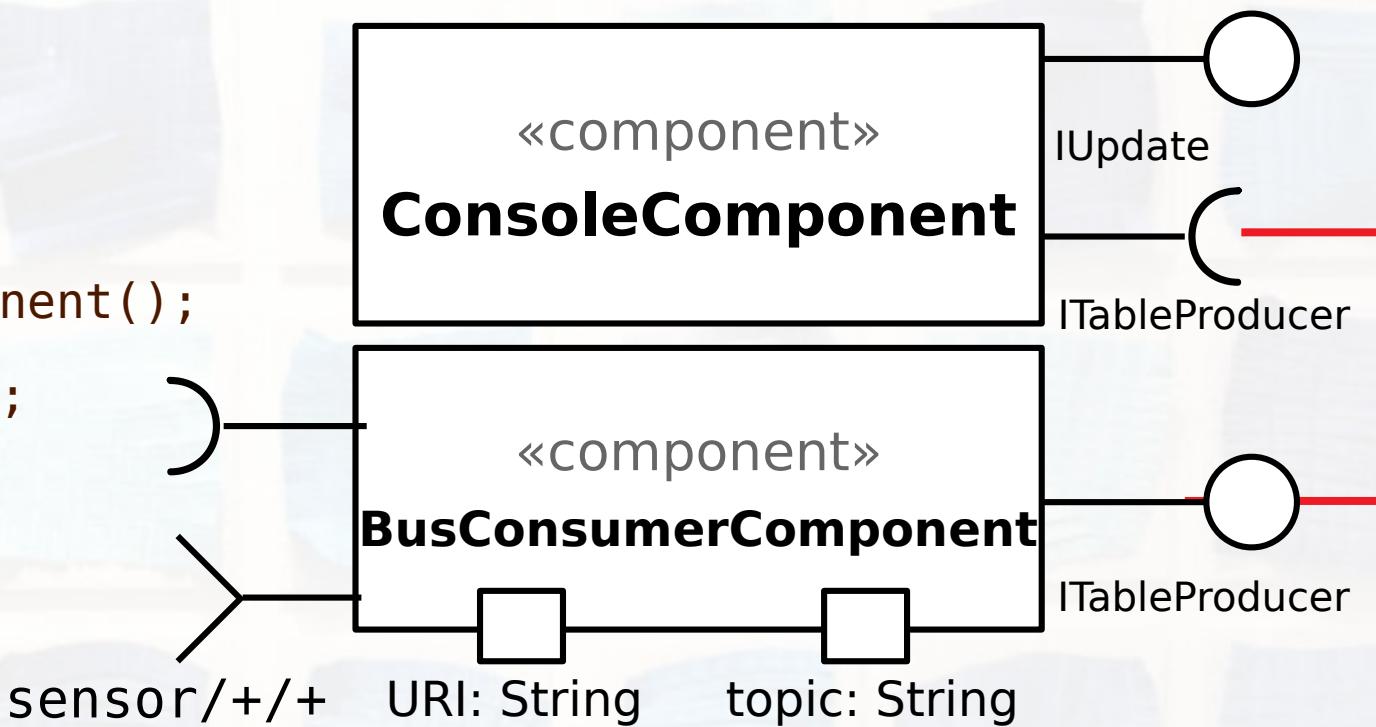
```
IConsole console =  
    new ConsoleComponent();
```



Conectando Componentes

```
IBusConsumer bc = new BusConsumerComponent();  
  
bc.setBusURI("tcp://localhost:1883");  
  
bc.setTopic("sensor/+/+");  
  
bc.setBlockSize(10);  
  
bc.setVerbose(2);
```

```
IConsole console =  
    new ConsoleComponent();  
  
console.connect(bc);
```



Conectando Componentes

```
IBusConsumer bc = new BusConsumerComponent();
```

```
bc.setBusURI("tcp://localhost:1883");
```

```
bc.setTopic("sensor/+/+");
```

```
bc.setBlockSize(10);
```

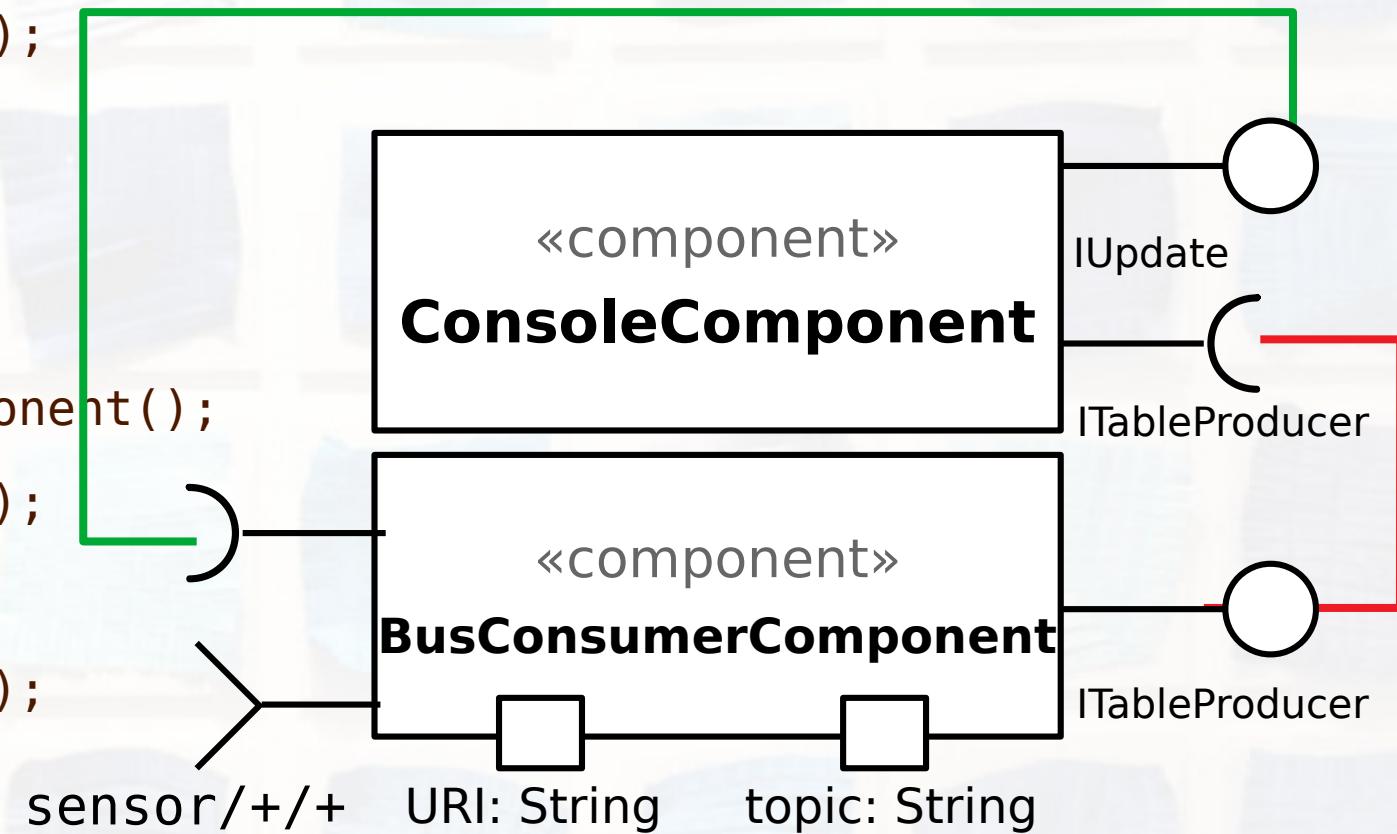
```
bc.setVerbose(2);
```

```
IConsole console =
```

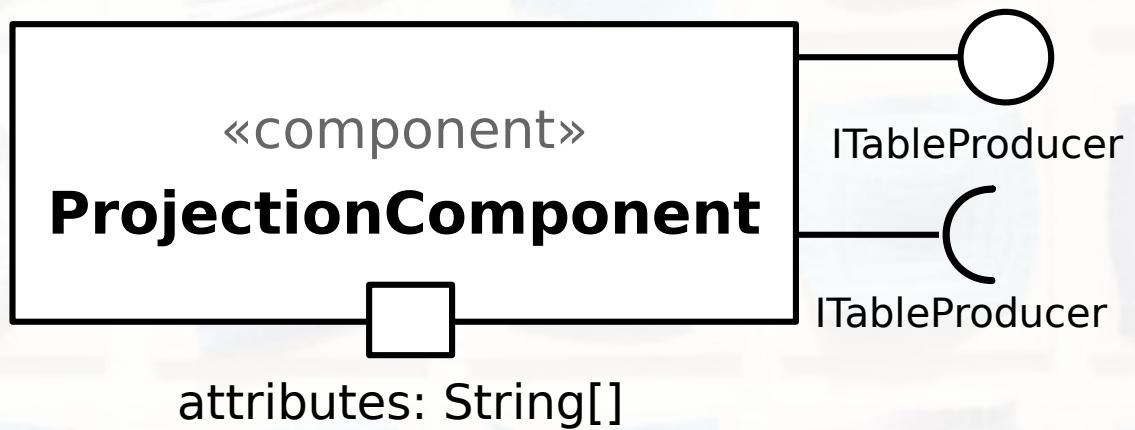
```
    new ConsoleComponent();
```

```
console.connect(bc);
```

```
bc.connect(console);
```

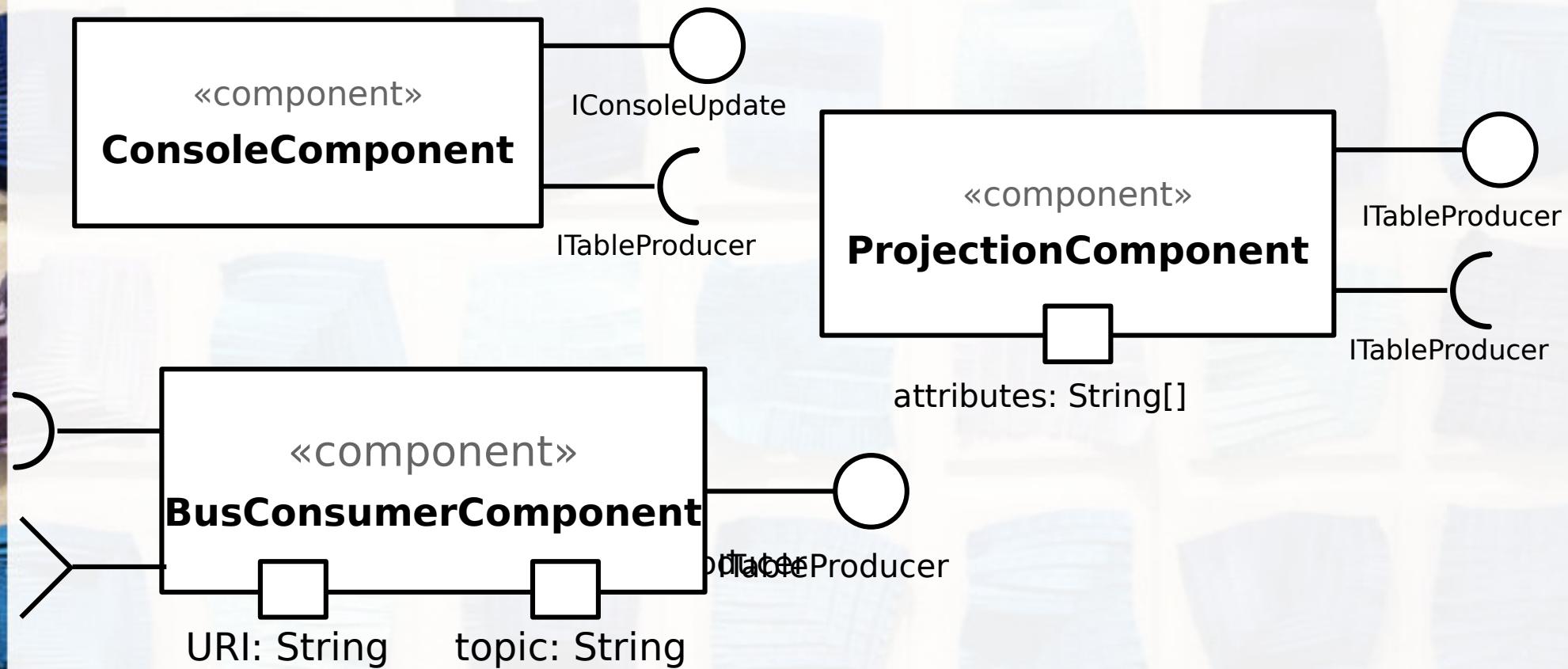


Componente Projection

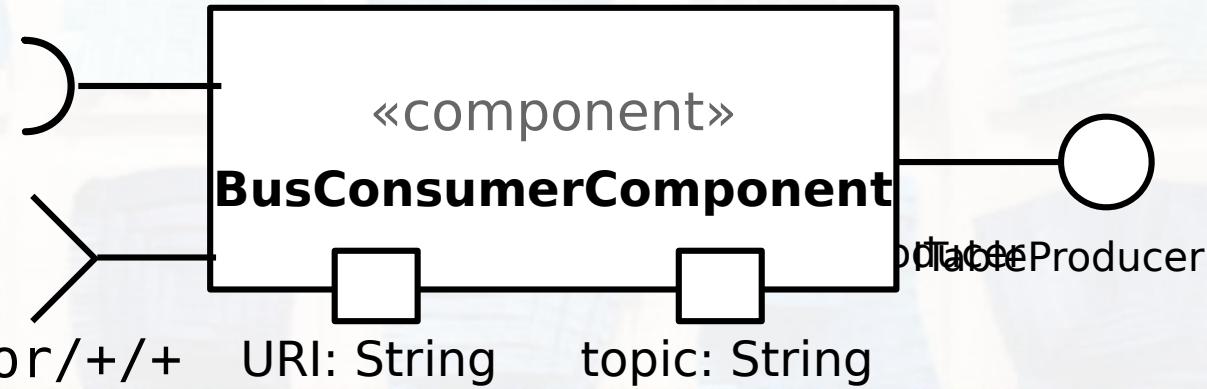
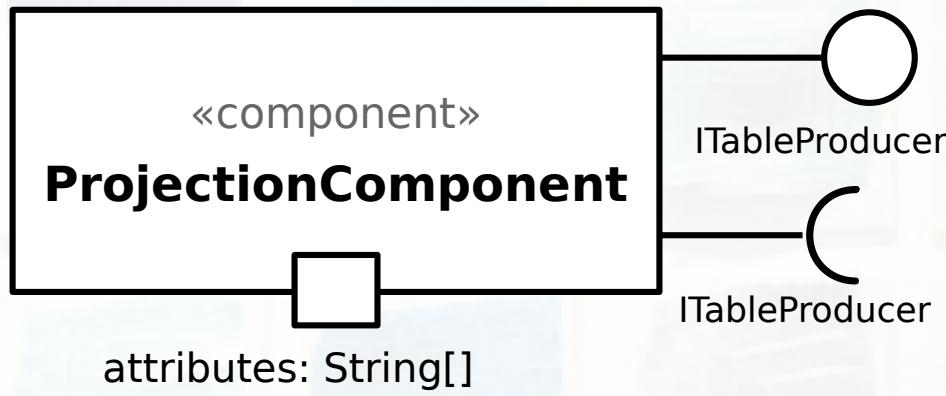
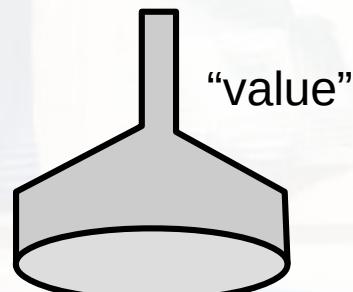
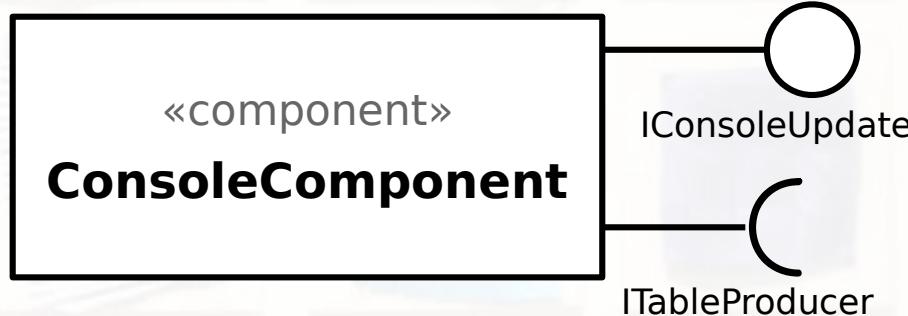
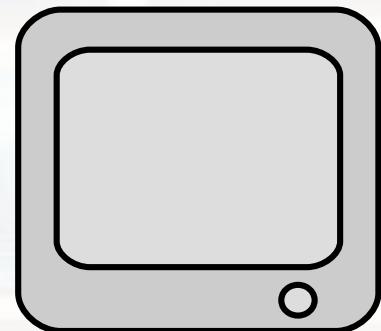


Exercício 3

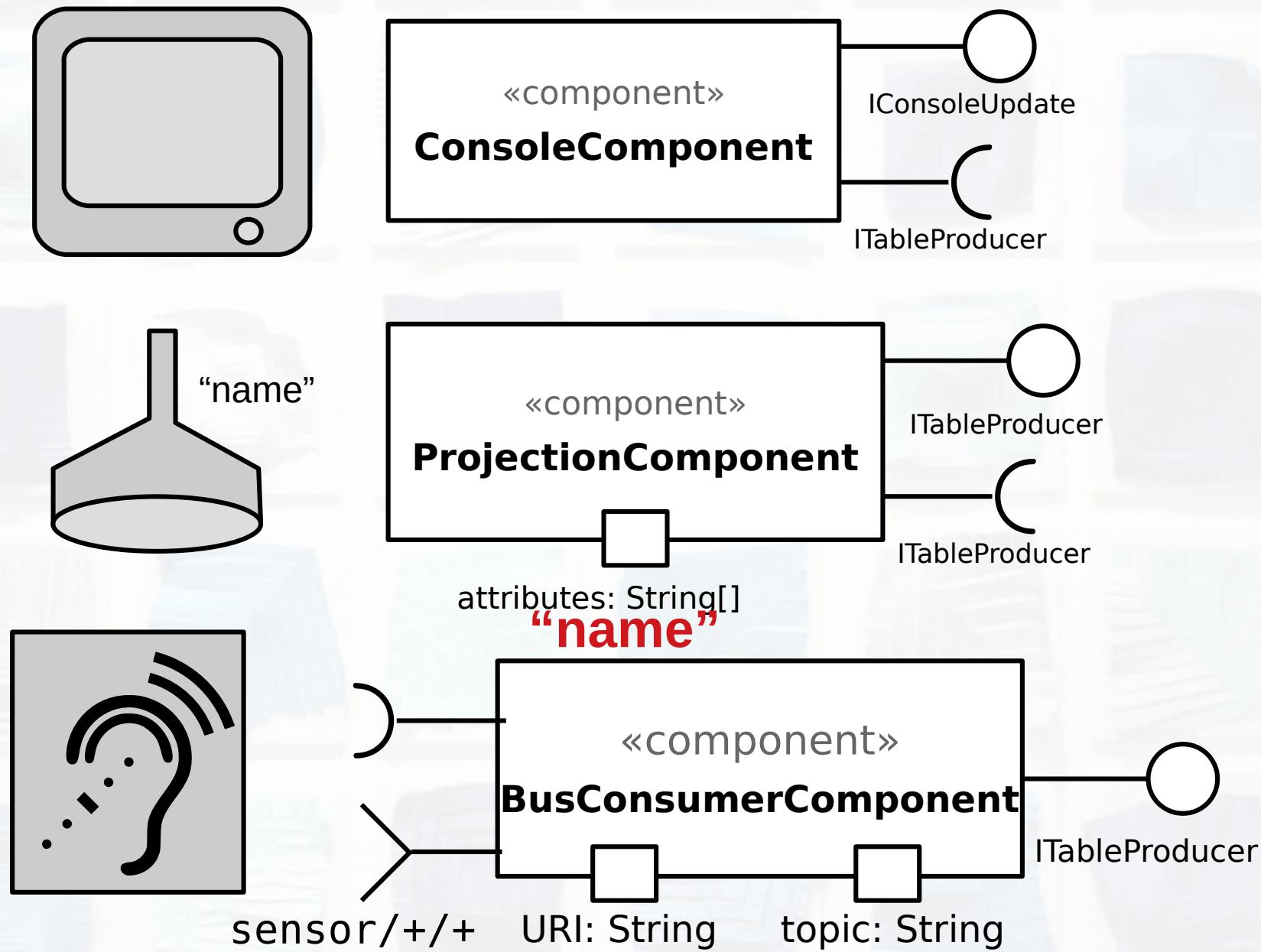
- Faça um diagrama UML de como seria a interface de um componente que realize uma filtragem da coluna de “value” da tabela.



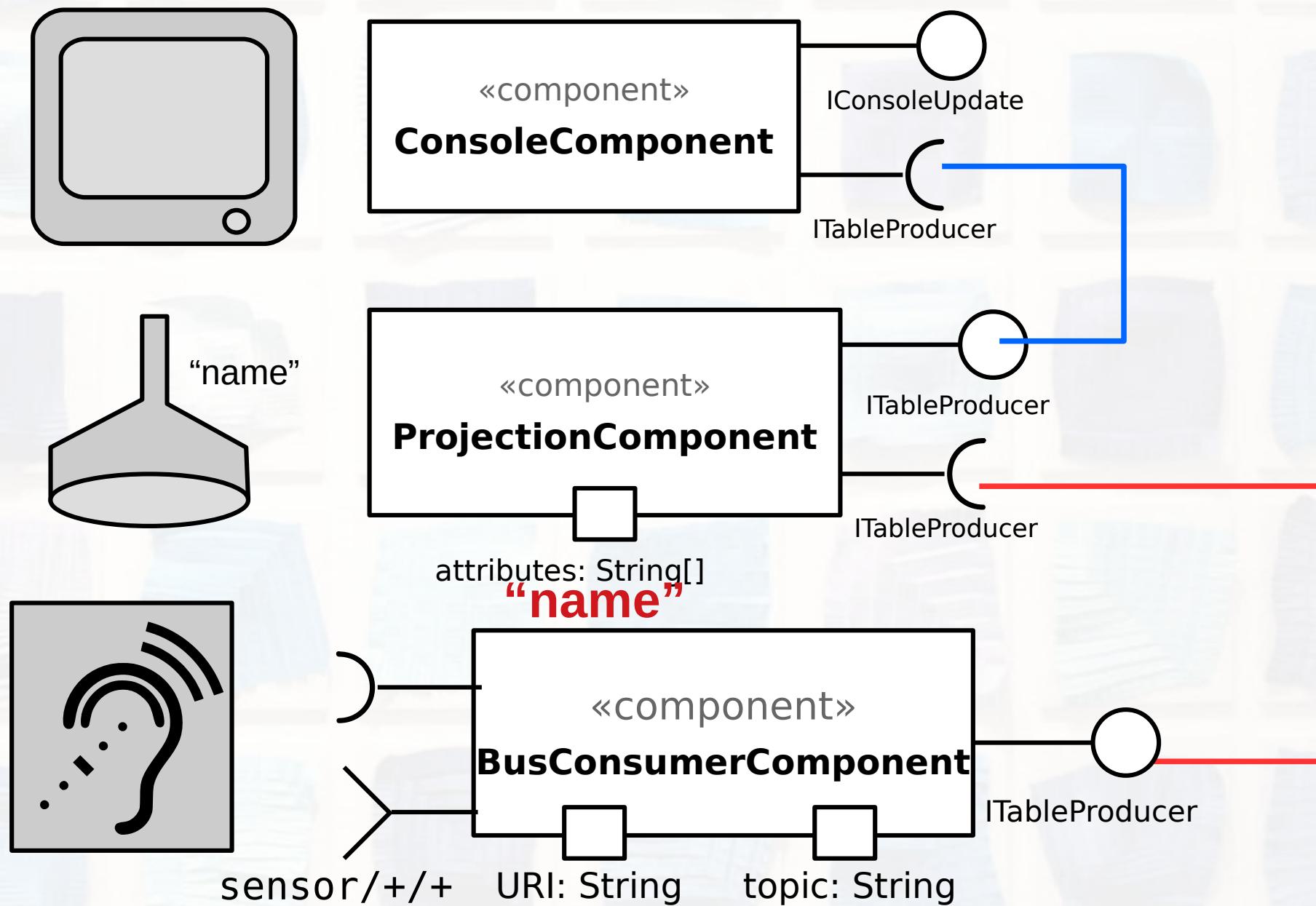
Conectando Três Componentes



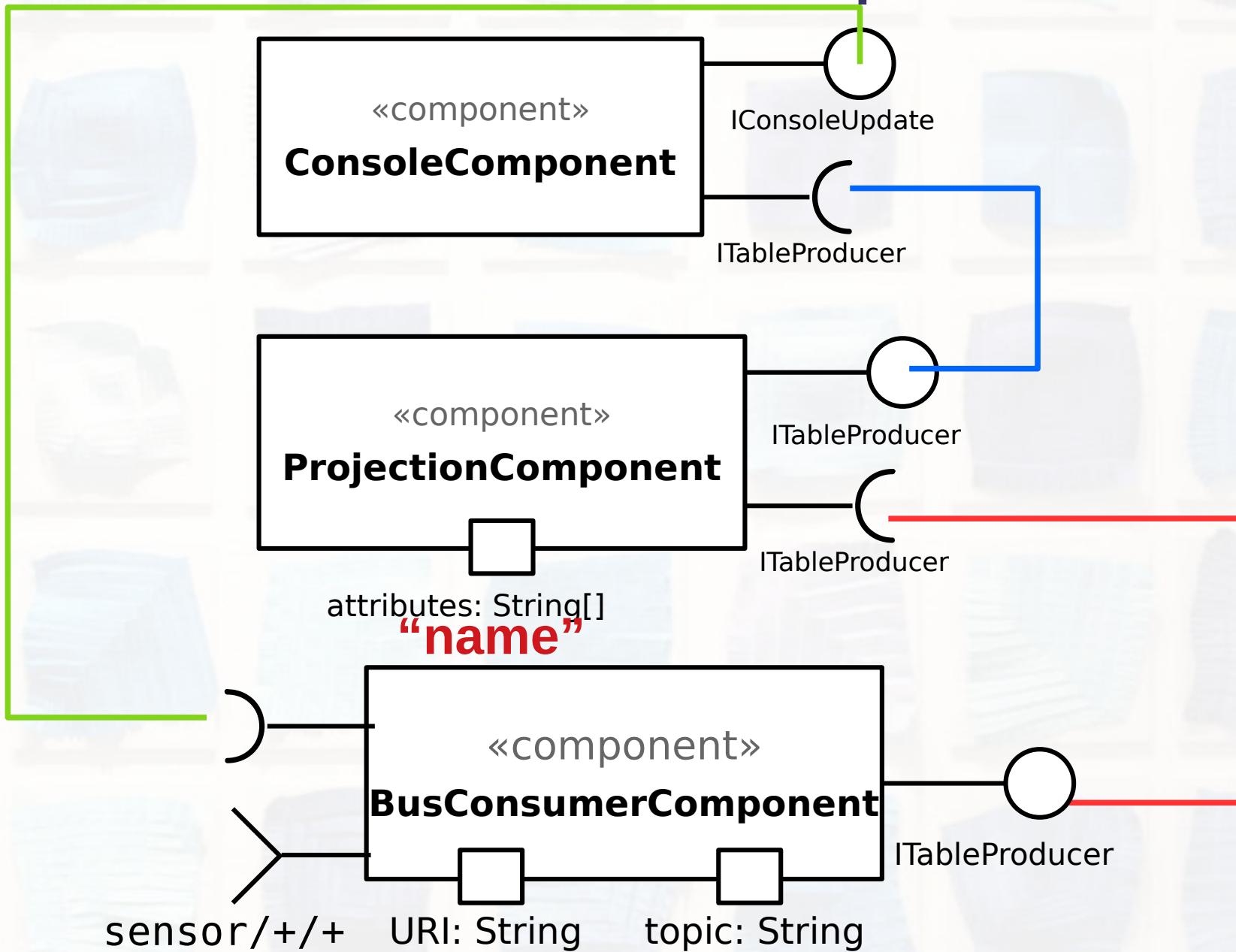
Conectando Três Componentes



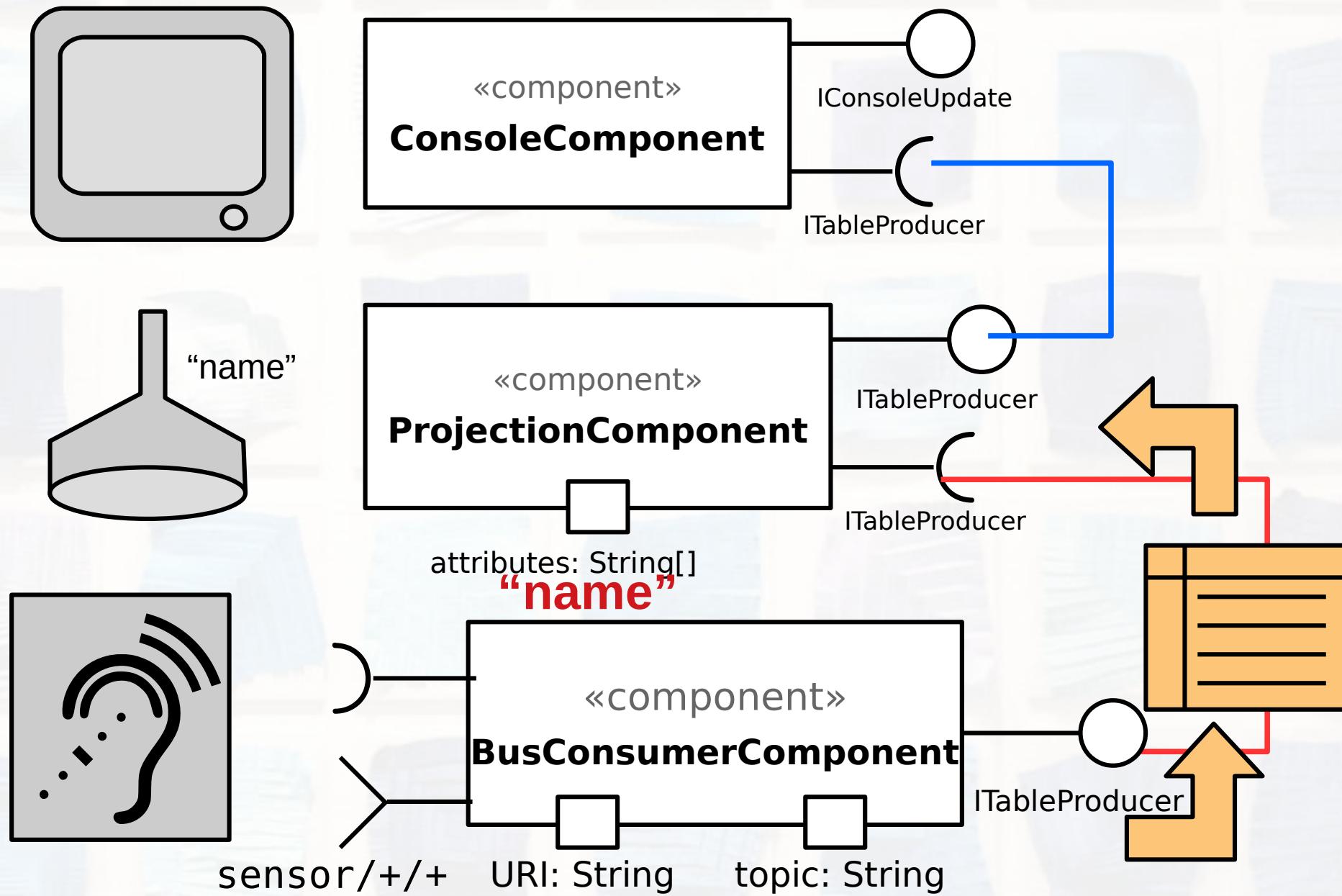
Conectando Três Componentes



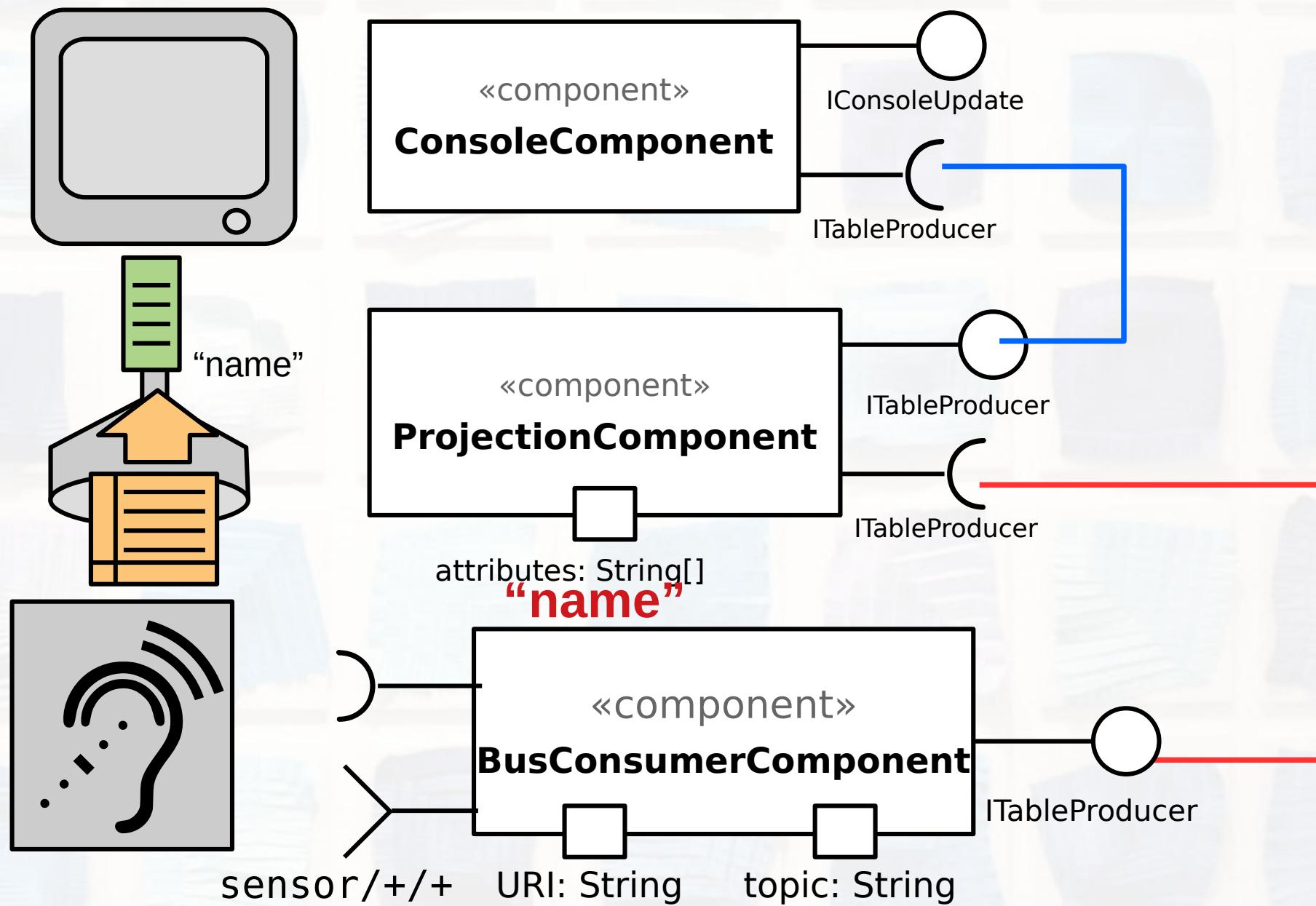
Conectando Três Componentes



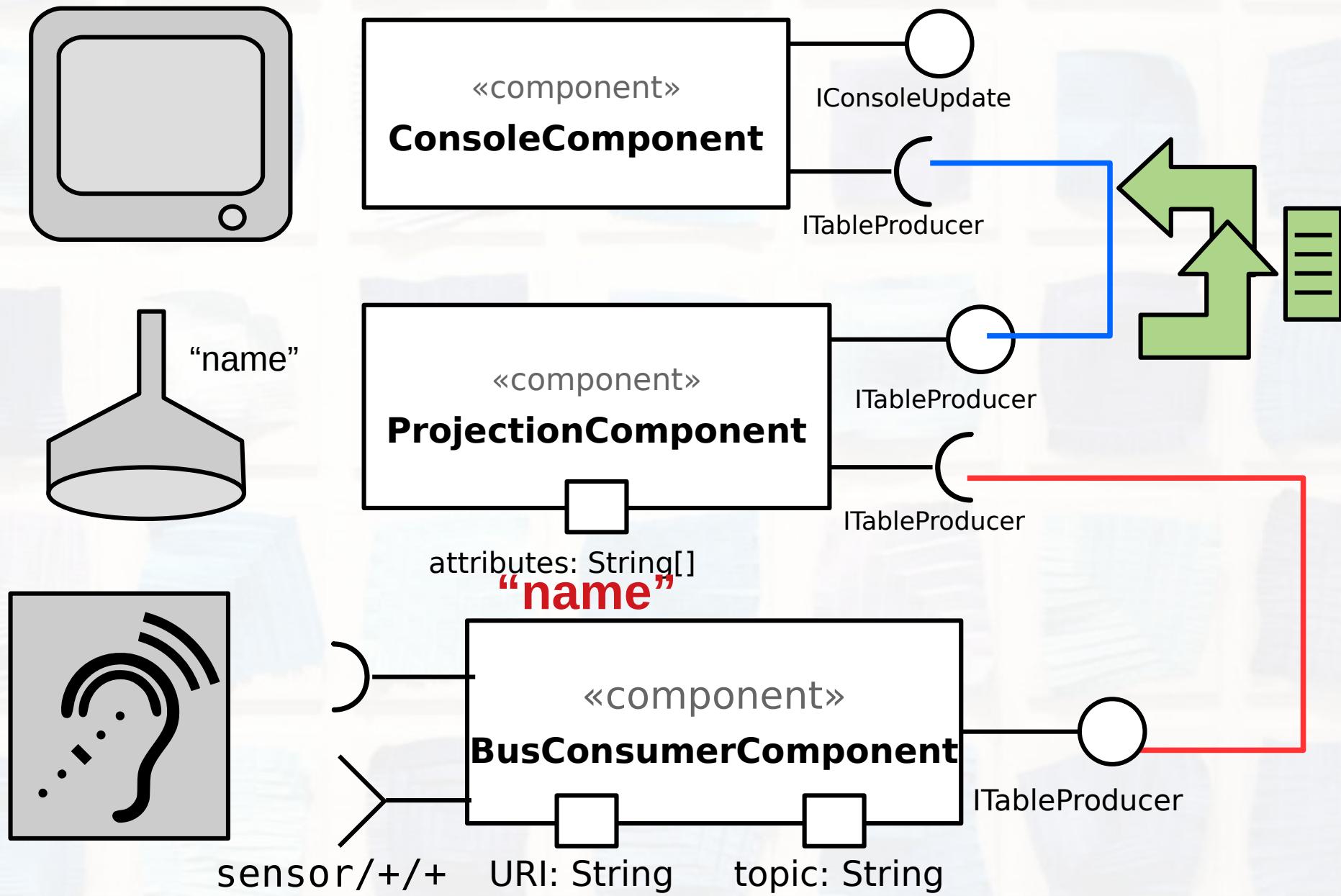
Conectando Três Componentes



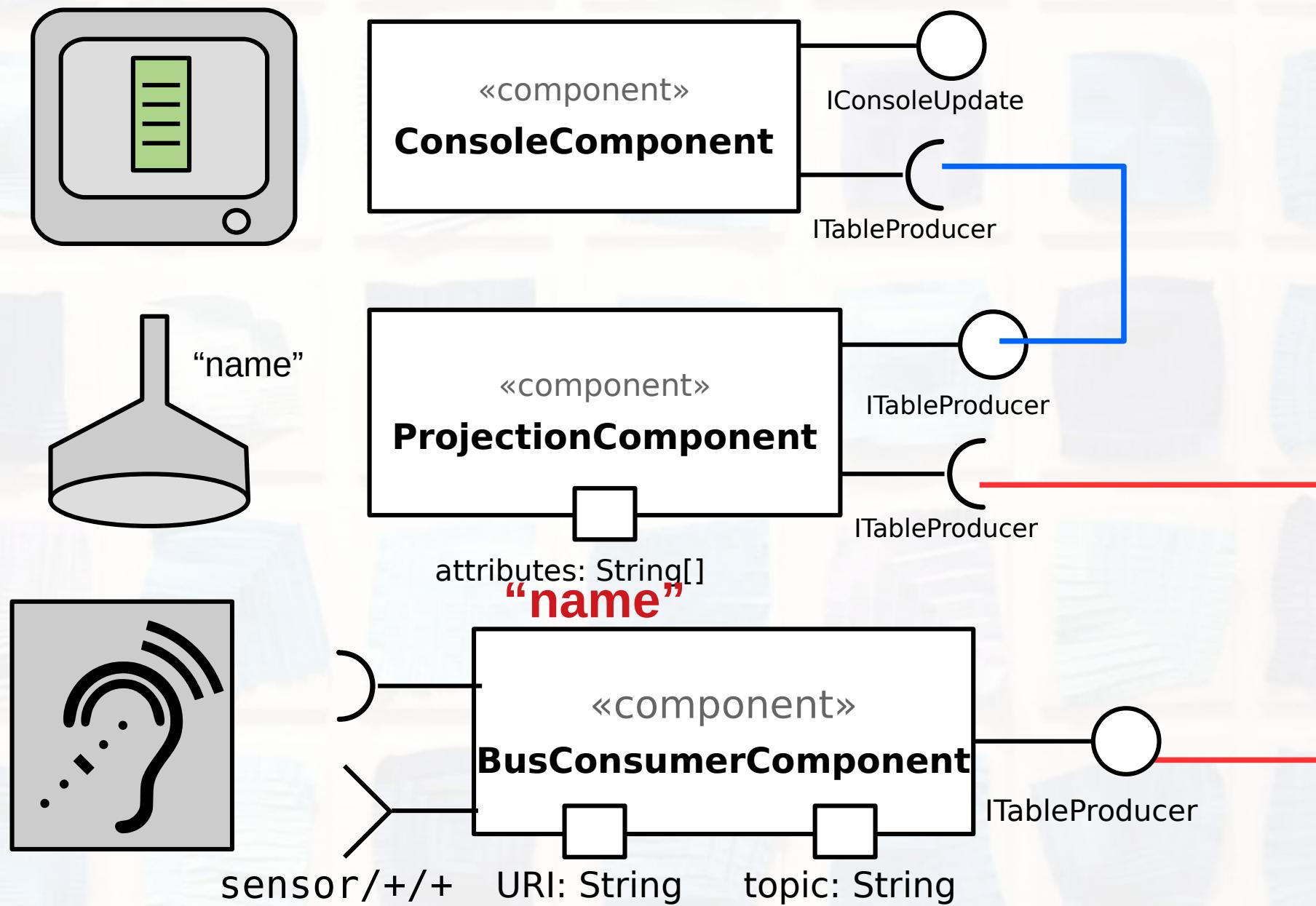
Conectando Três Componentes



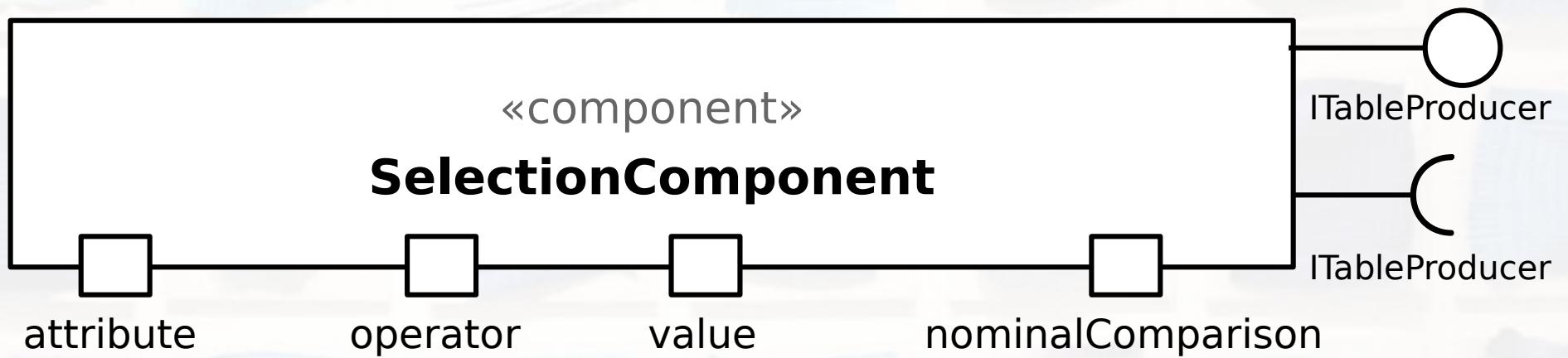
Conectando Três Componentes



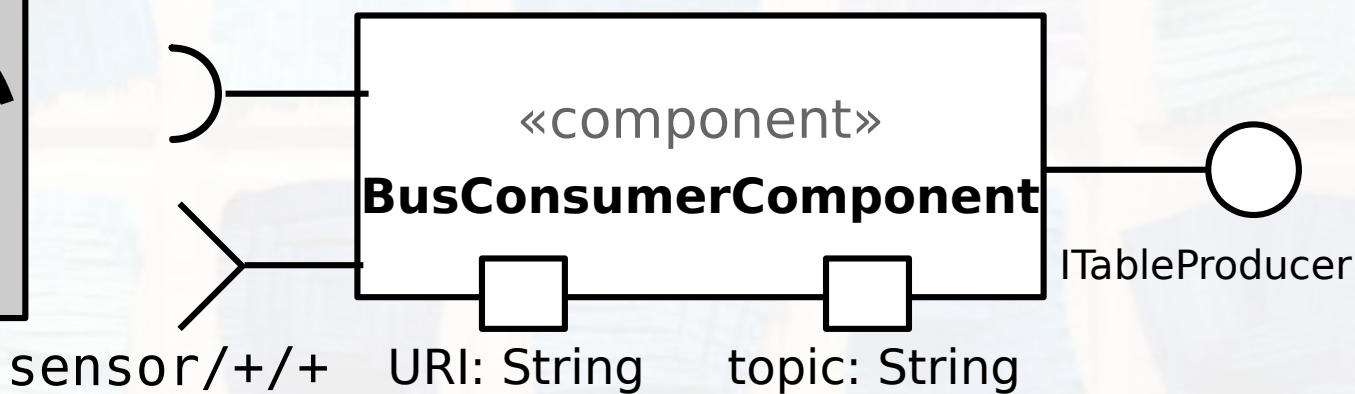
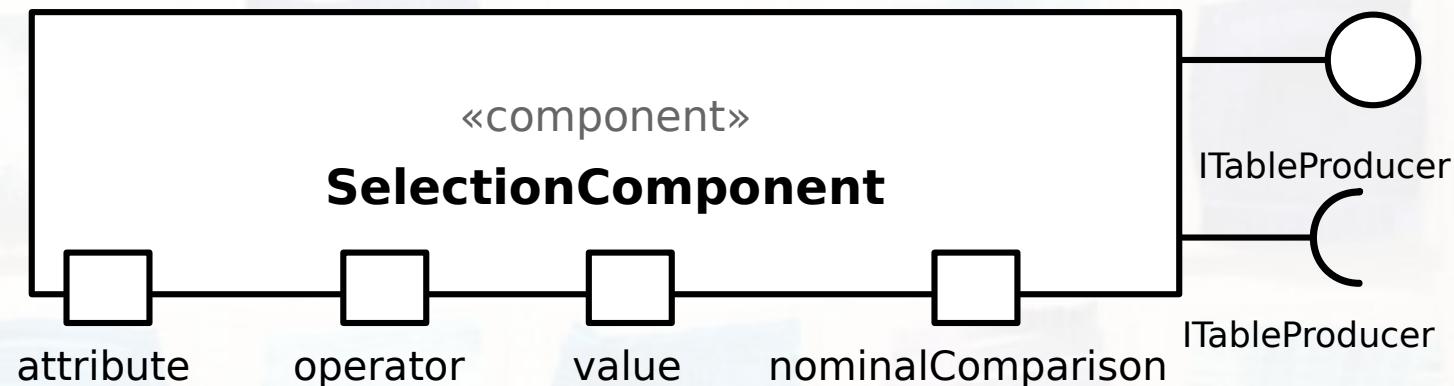
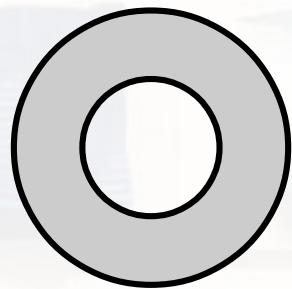
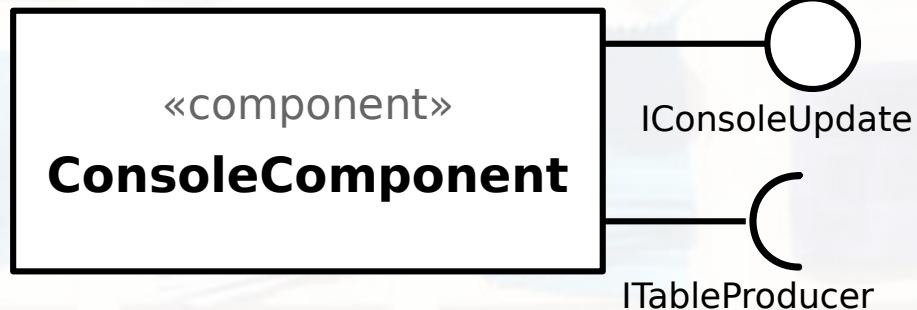
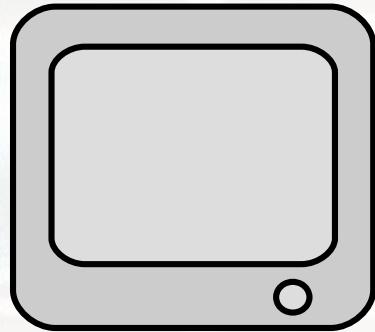
Conectando Três Componentes



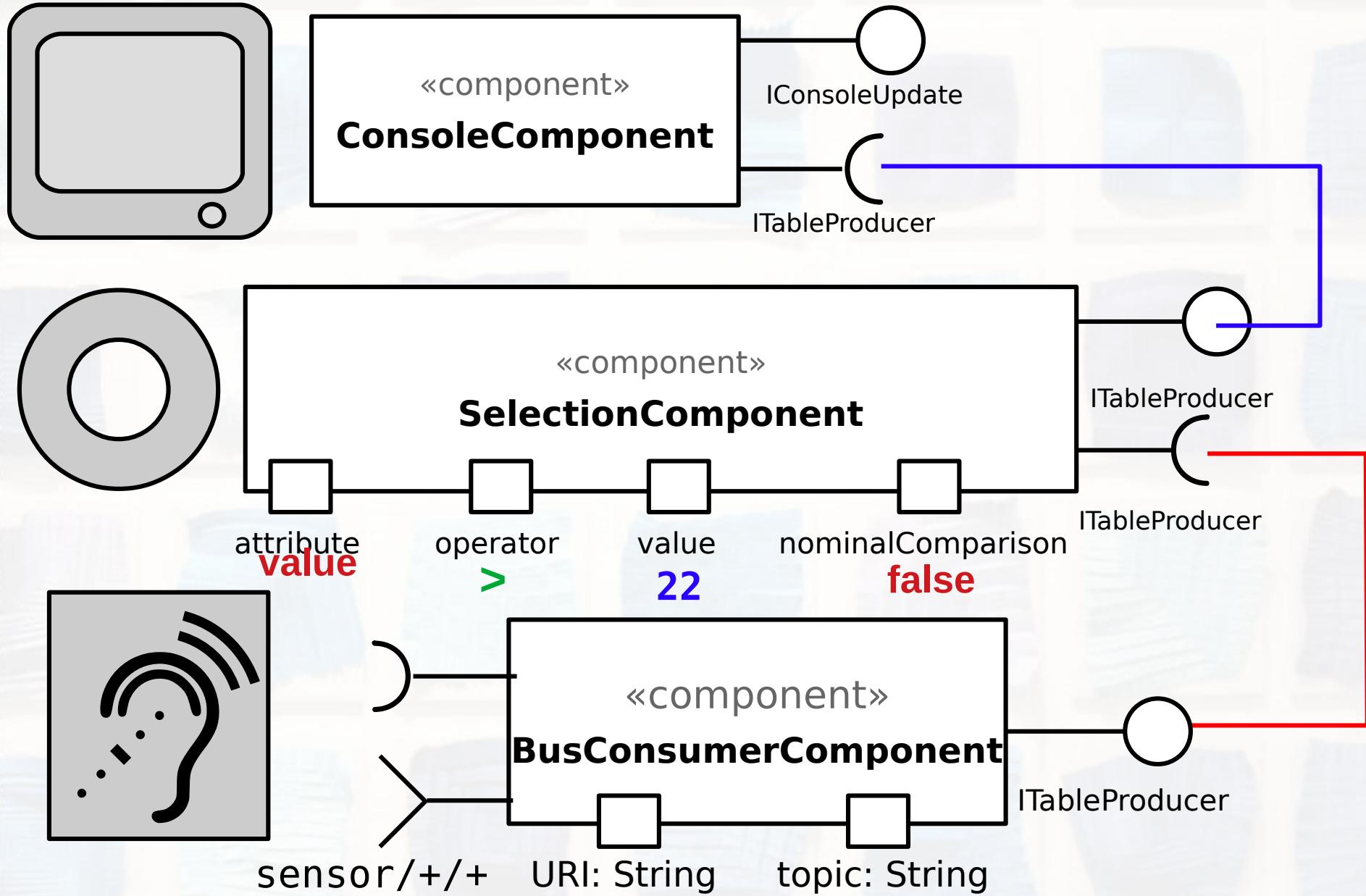
Componente Selection



Conectando com Selection



Conectando com Selection



Exercício 6

- Apresente um gráfico comparativo entre idade e tempo de recuperação, filtrando apenas a doença “bacterial_infection”.

Referências

- Agenda OpenSystems. COMPIERE – Smart Open Source ERP Software with integrated CRM Solutions. Disponível em <http://www.agenda.si/fileadmin/www.agenda.si/documents/Compiere.opis.pdf>, acessado em 9/04/2010.
- Ecma International (2011). ECMAScript Language Specification - Standard ECMA-262 (5.1 ed.).
- Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E. Gruber. 2008. Bigtable: A Distributed Storage System for Structured Data. ACM Trans. Comput. Syst. 26, 2, Article 4 (June 2008).

Referências

- Hickson, I. (2011). HTML Microdata -- W3C Working Draft 13 January 2011. W3C. Retrieved from <http://www.w3.org/TR/2011/WD-microdata-20110113/>
- Wolf, O. (2018). Introduction into Microservices. Retrieved August 20, 2018, from <https://specify.io/concepts/microservices>

André Santanchè
<http://www.ic.unicamp.br/~santanche>

Licença

- Estes slides são concedidos sob uma Licença Creative Commons. Sob as seguintes condições: Atribuição, Uso Não-Comercial e Compartilhamento pela mesma Licença.
- Mais detalhes sobre a referida licença Creative Commons veja no link:
https://creativecommons.org/licenses/by-nc-sa/4.0/deed.pt_BR
- Agradecimento a Goran Konjevod [<https://www.flickr.com/photos/23913057@N05/>] por sua fotografia “50 waves” usada na capa e nos fundos, disponível em [<https://flic.kr/p/8msVPU>] vide licença específica da fotografia.