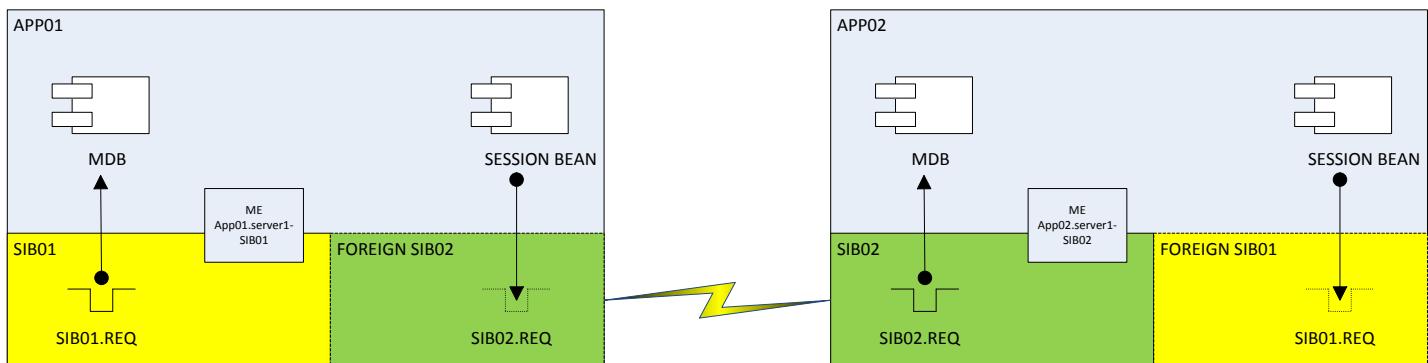


## Índice

<b>1. Exercício X – Configurar o Service Integration Bus.....</b>	<b>2</b>
<b>1.1. Arquitetura Ilustrativa .....</b>	<b>2</b>
<b>2. Configuração do Nó App01.....</b>	<b>3</b>
<b>2.1. Configuração do Service Integration Bus SIB01 .....</b>	<b>3</b>
<b>2.2. Criar um Bus Member para o Service Integration Bus .....</b>	<b>6</b>
<b>2.3. Criar um Foreign Service Integration Bus .....</b>	<b>12</b>
<b>2.4. Criar uma Local Queue e uma Foreign Queue .....</b>	<b>17</b>
<b>2.5. Criar filas JMS.....</b>	<b>23</b>
<b>2.6. Criar um Activation Spec .....</b>	<b>27</b>
<b>2.7. Criar um Queue Connection Factory .....</b>	<b>30</b>
<b>3. Configuração do Nó App02.....</b>	<b>33</b>
<b>3.1. Configuração do Service Integration Bus SIB02.....</b>	<b>33</b>
<b>3.2. Criar um Bus Member para o Service Integration Bus .....</b>	<b>34</b>
<b>3.3. Criar um Foreign Service Integration Bus .....</b>	<b>35</b>
<b>3.4. Criar uma Local Queue e uma Foreign Queue .....</b>	<b>36</b>
<b>3.5. Criar filas JMS.....</b>	<b>37</b>
<b>3.6. Criar um Activation Spec .....</b>	<b>38</b>
<b>3.7. Criar um Queue Connection Factory .....</b>	<b>39</b>
<b>4. Desenvolvimento da aplicação de teste da mensageria.....</b>	<b>40</b>
<b>4.1. Construção da aplicação de teste de mensageria (BVMessaging.EAR) .....</b>	<b>40</b>
<b>5. Instalação da aplicação de teste da mensageria.....</b>	<b>52</b>
<b>5.1. Instalação da aplicação de teste (BVMessaging.EAR) .....</b>	<b>52</b>
<b>5.2. Testes da aplicação.....</b>	<b>58</b>

## 1. Exercício X – Configurar o Service Integration Bus

### 1.1. Arquitetura Illustrativa



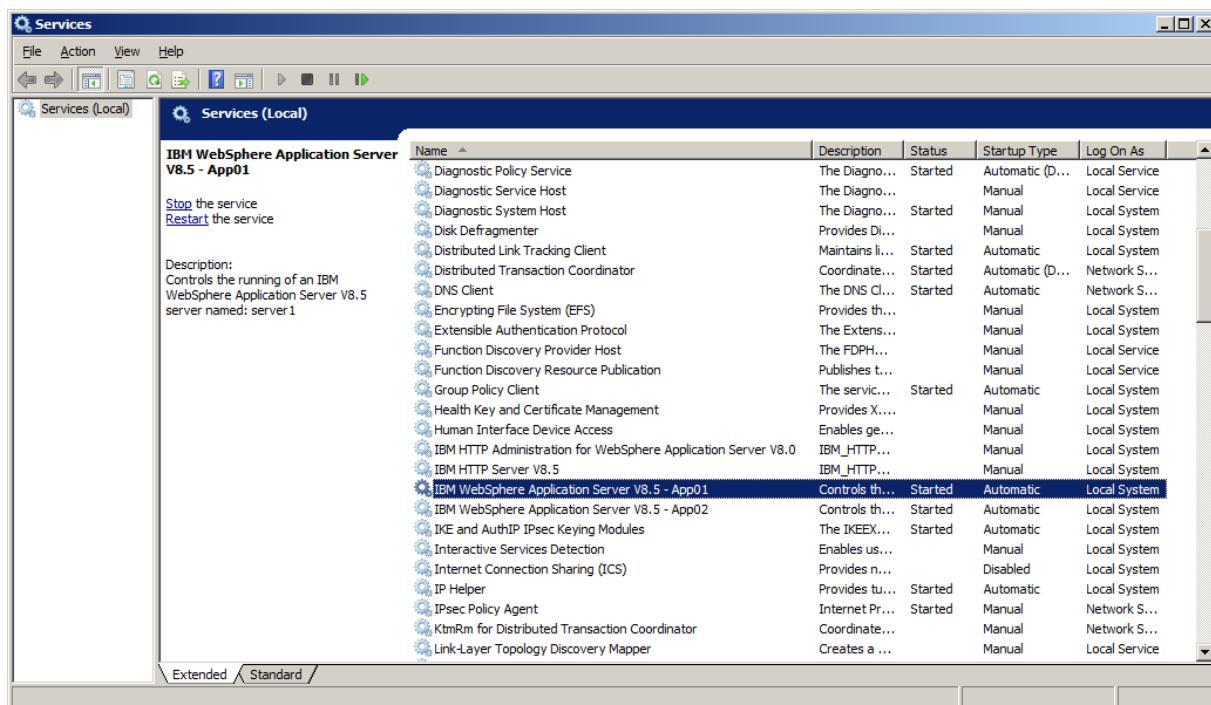
O cenário de integração é composto por dois servidores de aplicação. Cada um terá:

- Um Service Integration Bus (ex. SIB01);
- Um Bus Member e um Messaging Engine (ex. App01.server1- SIB01);
- Um Foreign Service Integration Bus integrado ao Service Integration Bus do outro servidor de aplicação (ex. SIB02);
- Uma Local Queue (ex. SIB01.REQ)
- Uma Foreign Queue (ex. SIB02.REQ);
- Uma fila JMS para consumir as mensagens (ex. jms/SIB01.REQ);
- Um Activation Spec (ex. jms/APPActivationSpec) para o MDB consumir as mensagens da fila;
- Uma fila JMS para postar mensagens (ex. jms/REQ);
- Uma Queue Connection Factory (ex. jms/QCF) para o Session Bean postar mensagens na fila de requisição;

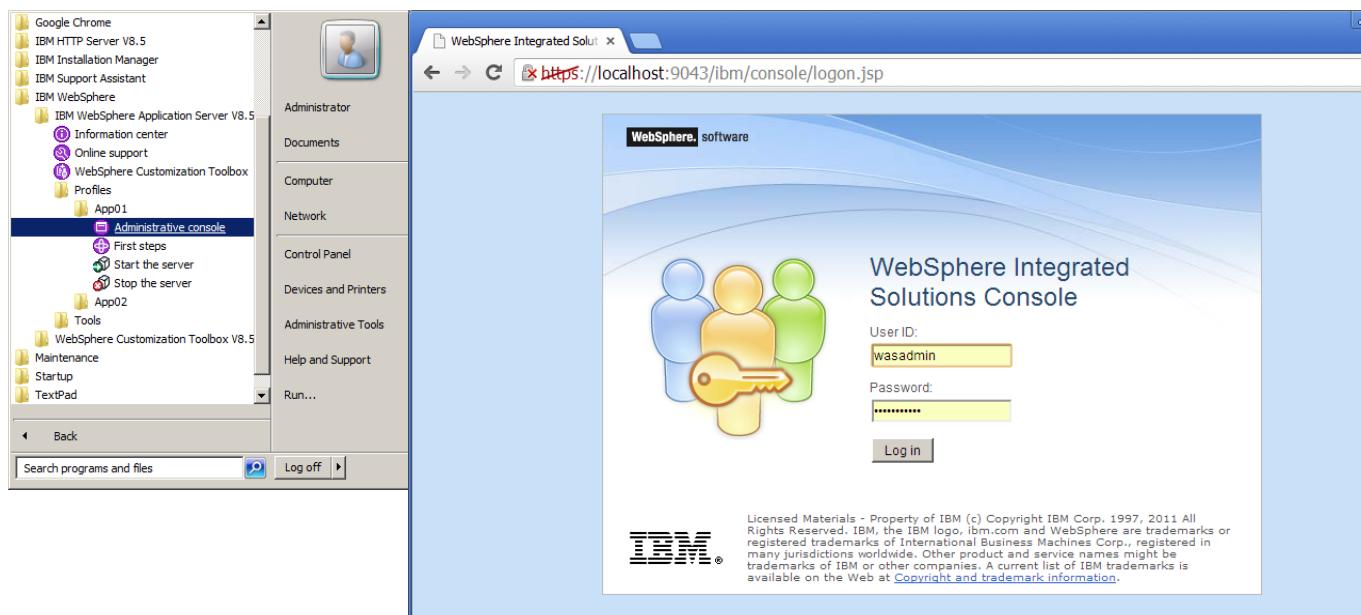
## 2. Configuração do Nô App01

### 2.1. Configuração do Service Integration Bus SIB01

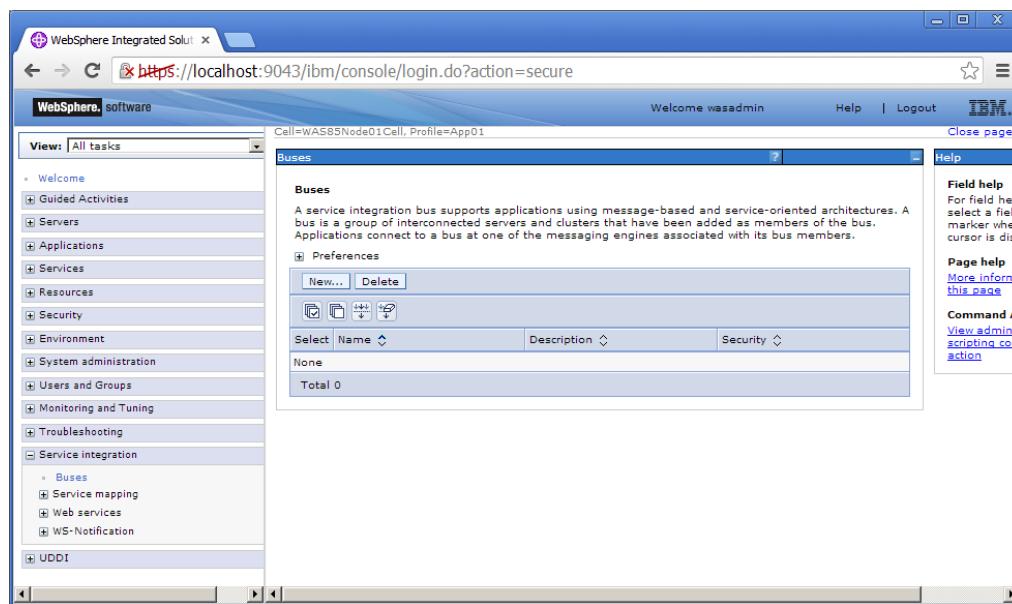
- j) Abra os serviços do Windows e verifique se os serviços referentes aos servidores de aplicação estão no ar:



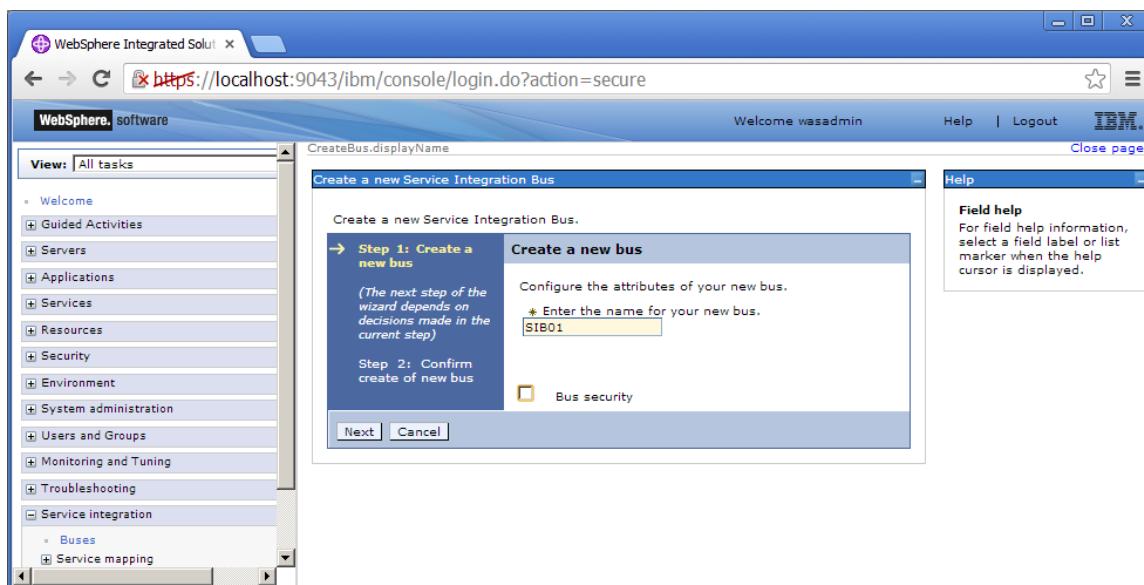
- k) Abra a console administrativa do App01 (<http://localhost:9060/admin>) ou pelo atalho:



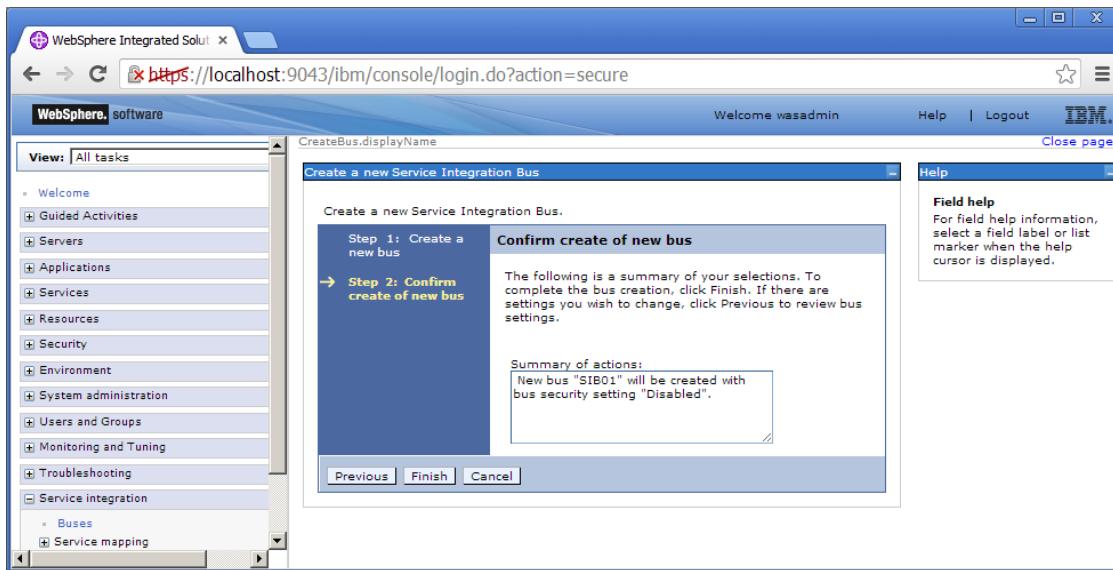
- I) Digite o usuário wasadmin / Cert0104sys e clique Login. No menu a esquerda, expanda Service Integration e clique em Buses:



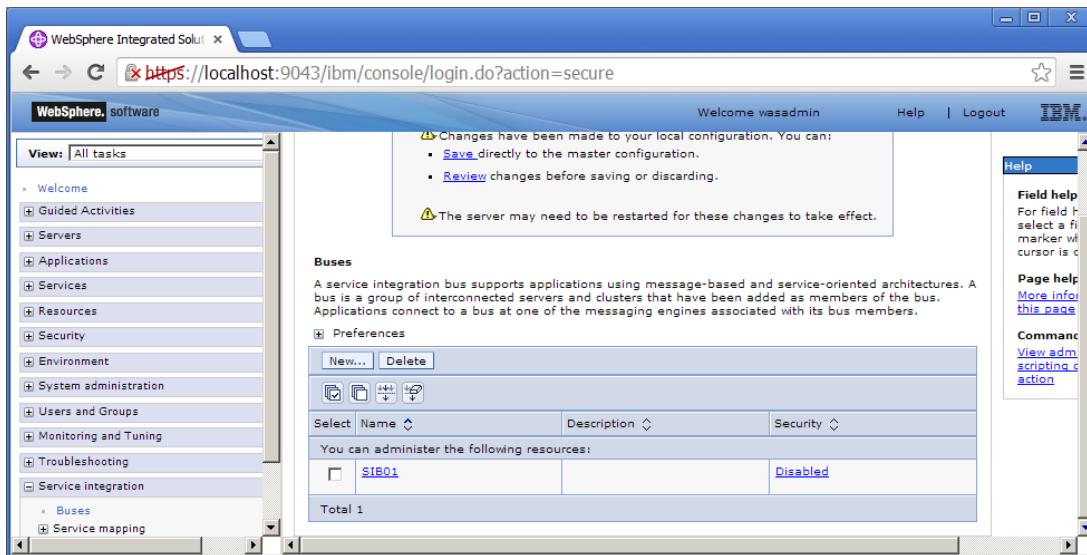
- m) Clique New. Digite SIB01 em Enter the name for your new bus e DESMARQUE a opção Bus security:



n) Clique Next. Reveja as configurações feitas:



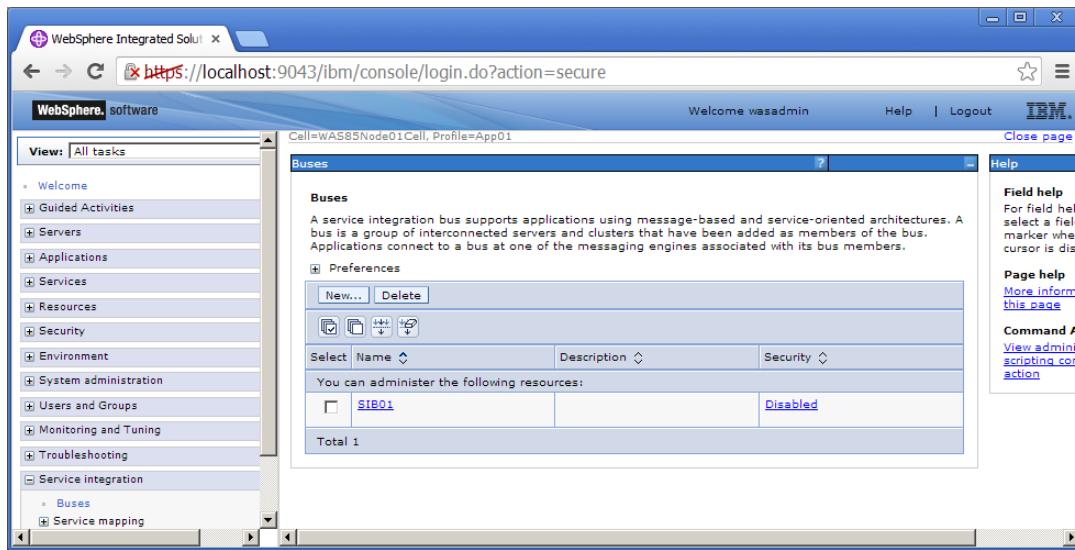
o) Clique Finish para criar o Service Integration Bus SIB01:



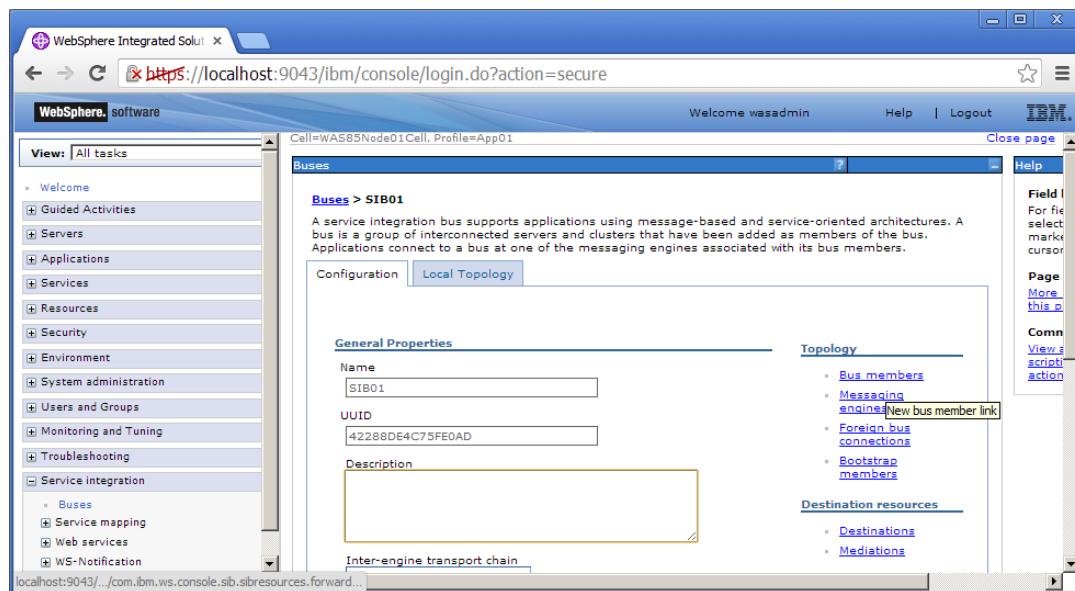
p) Salve as configurações na Master Configuration.

### 2.2. Criar um Bus Member para o Service Integration Bus

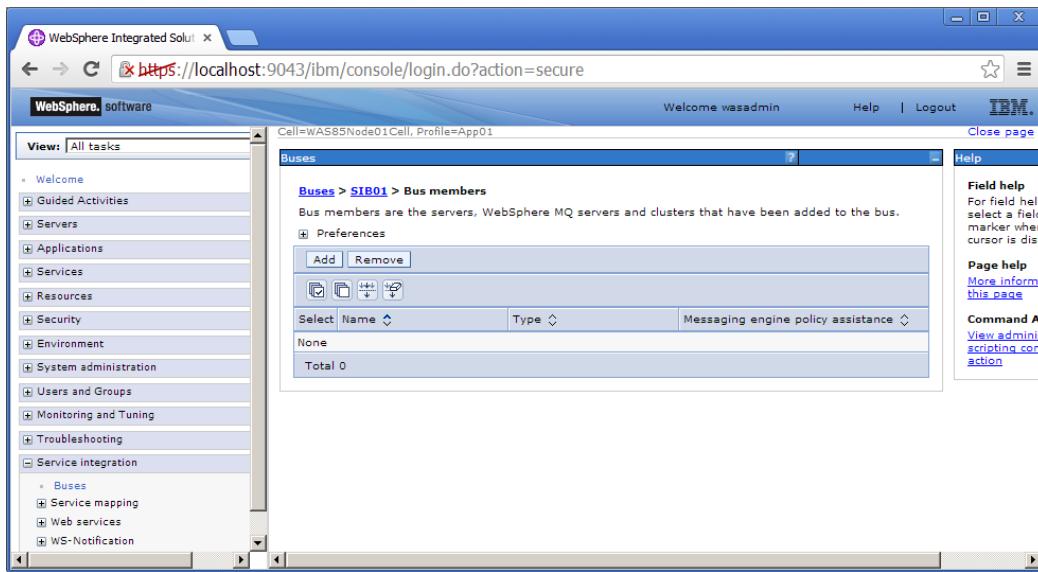
- Na console, expanda Service Integration e clique em Buses:



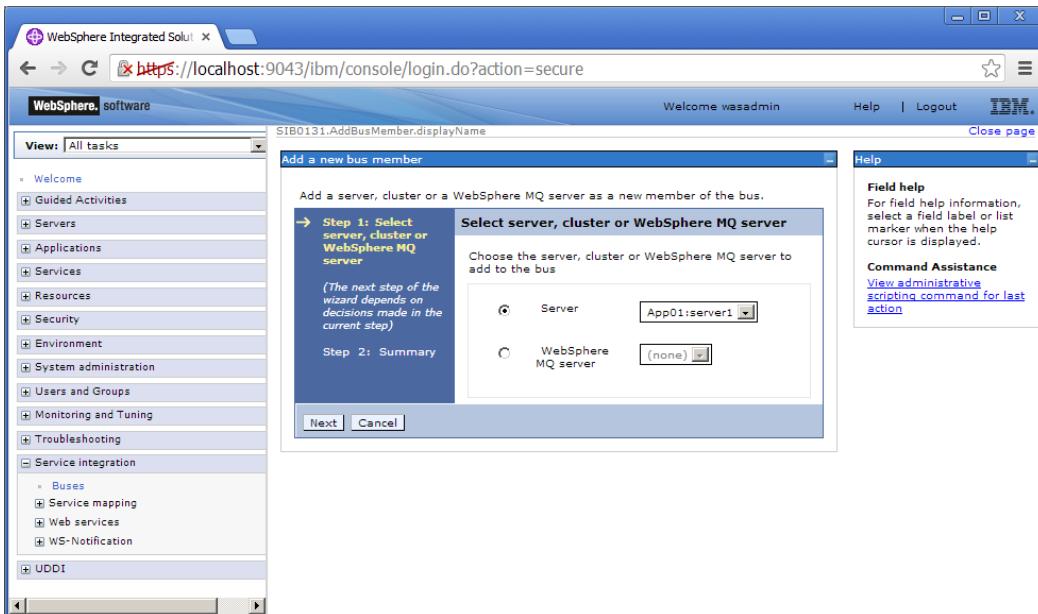
- Clique em SIB01:



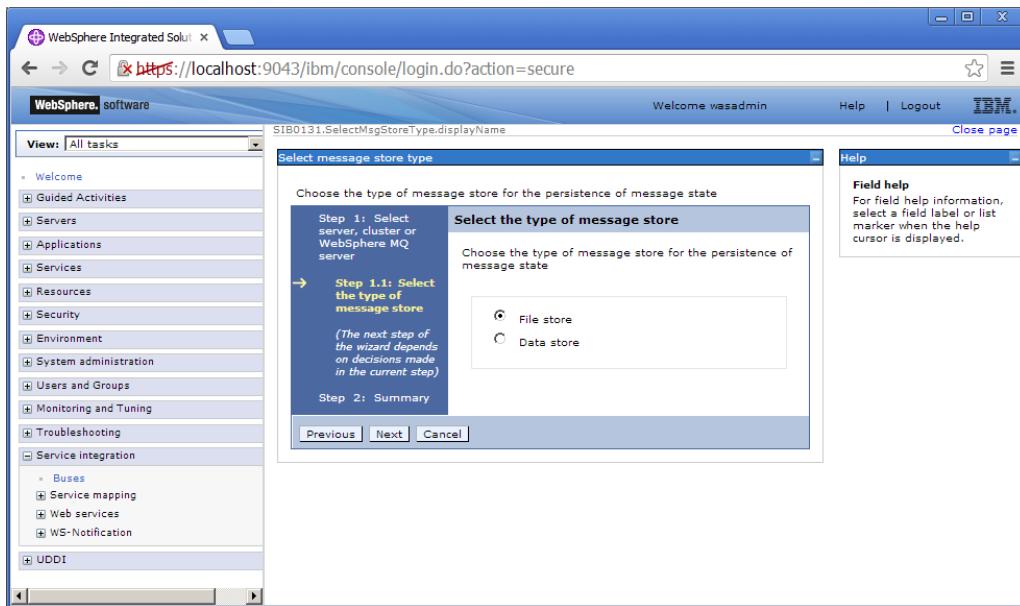
c) Clique em Topology-> Bus Members:



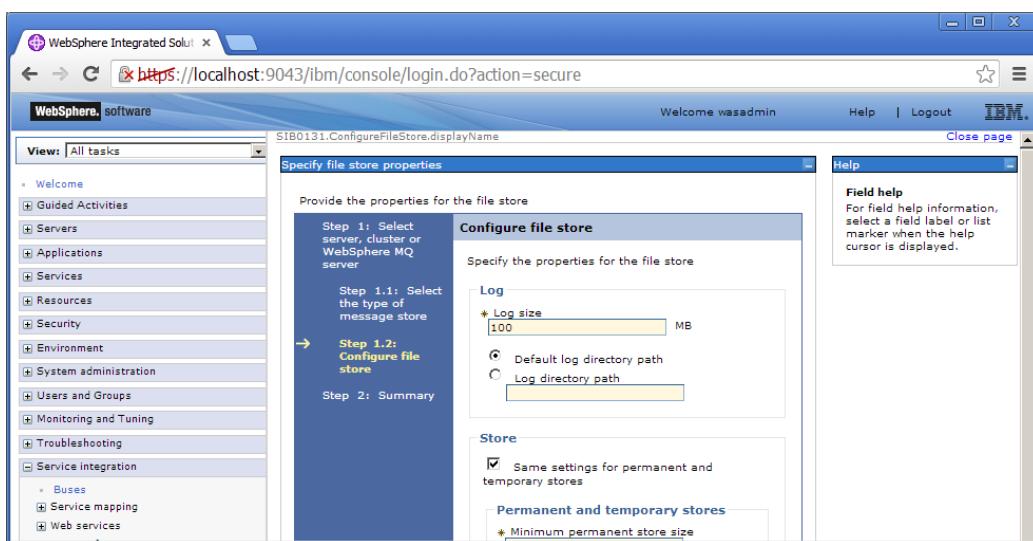
d) Clique em Add para adicionar um Bus Member ao Service Integration Bus:



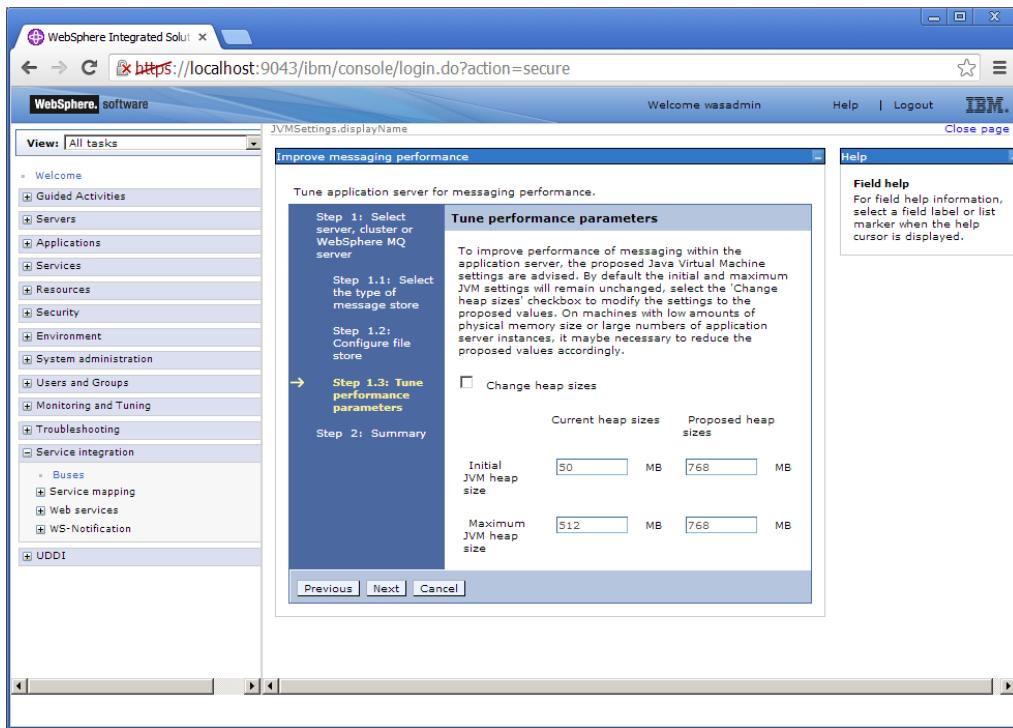
e) Selecione Server (App01:server1) e clique Next:



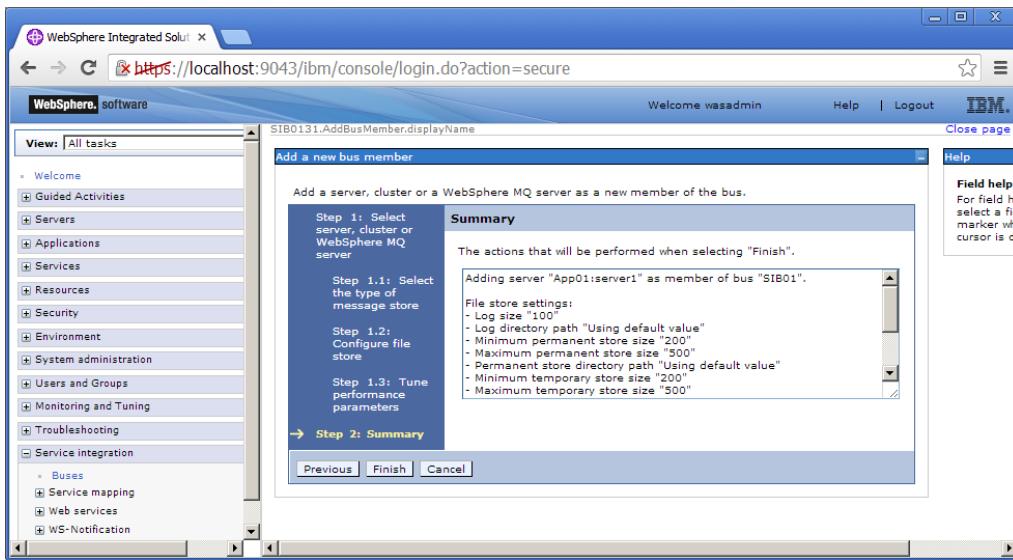
f) Selecione File Store e Clique Next:



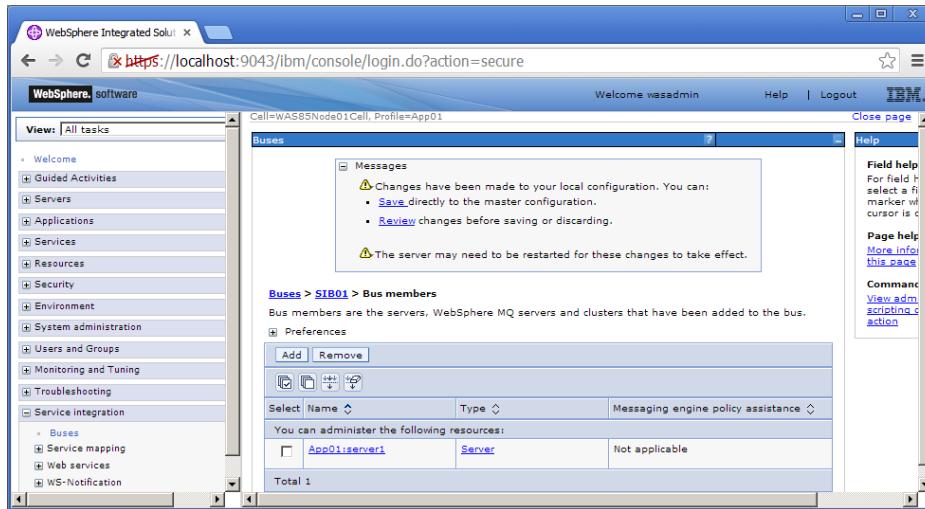
g) Mantenha as configurações padrões e clique Next:



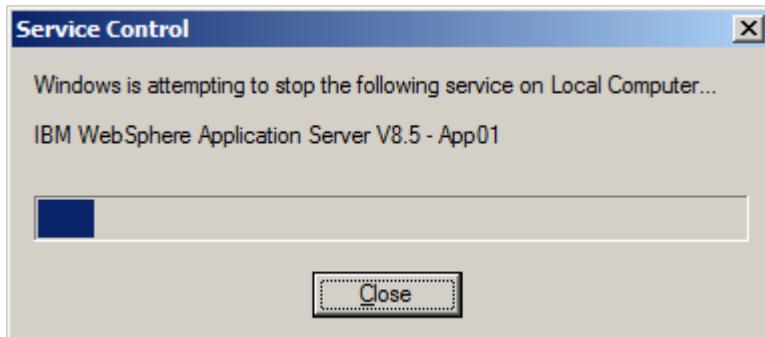
h) Mantenha as configurações padrões e clique Next:



- i) Revise as configurações realizadas e clique Finish:

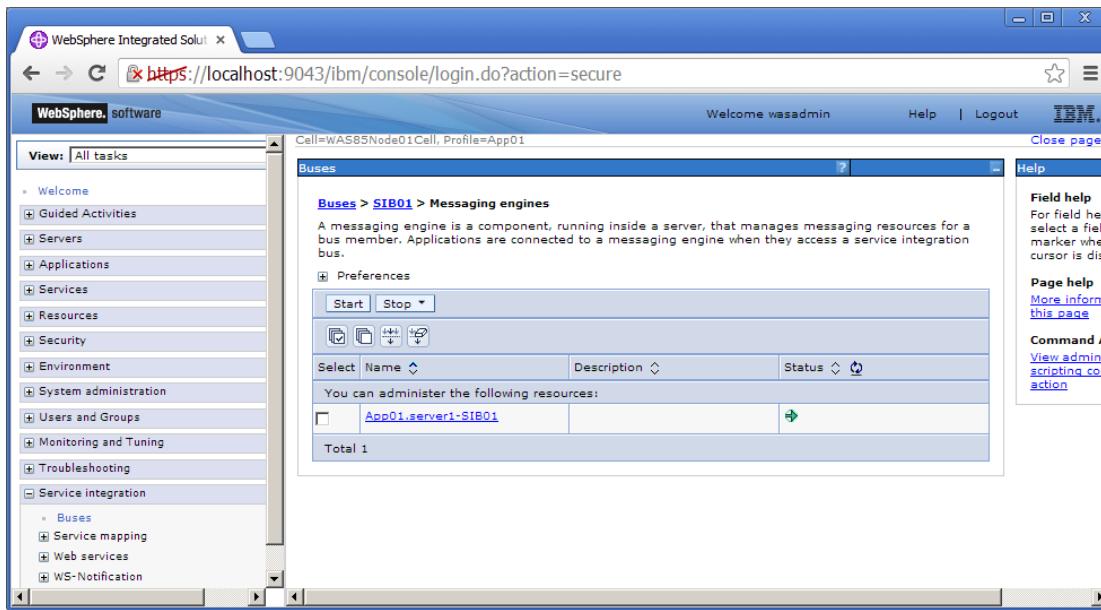


- j) Salve as configurações feitas na Master Configuration. Faça um logout da console.  
k) Nos serviços do Windows, reinicie o serviço IBM WebSphere Application Server V8.5 - App01:



- l) Entre novamente na Console Administrativa.  
m) Expanda Service Integration e Clique em Buses.  
n) Clique em SIB01.

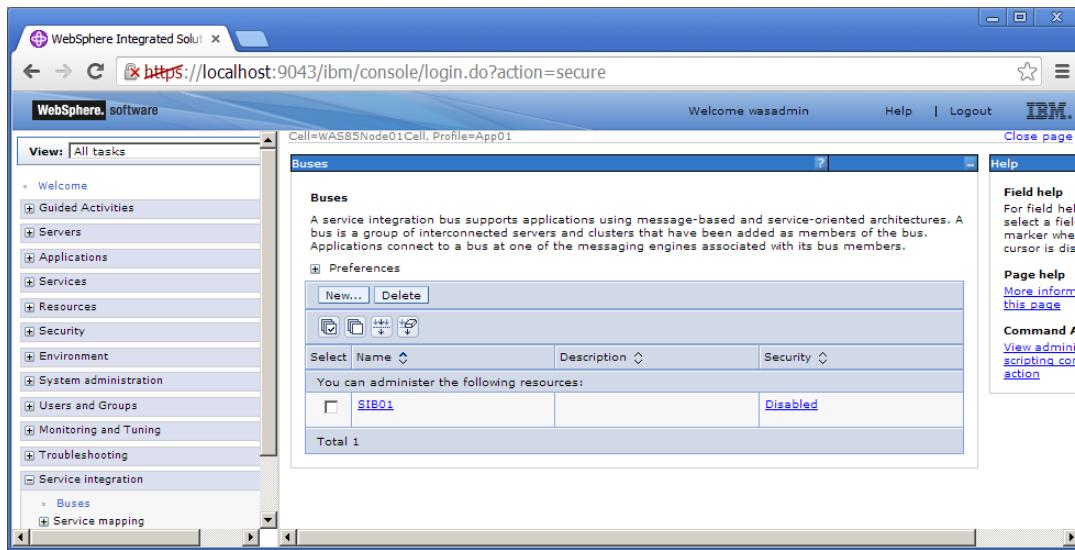
o) Clique em Topology->Messaging Engines:



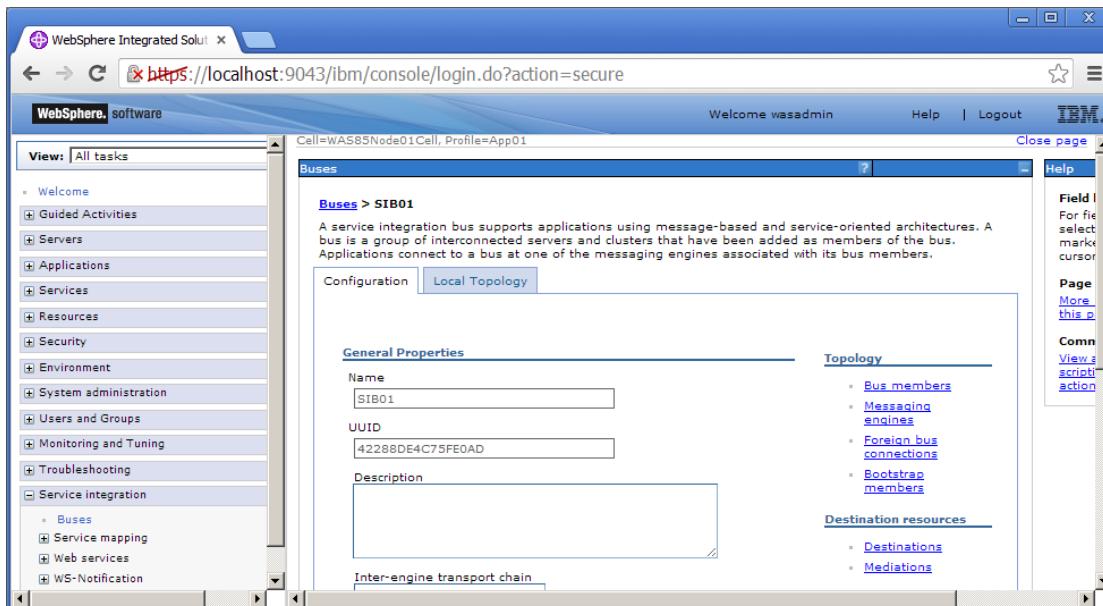
p) Verifique que o Messaging Engine App01.server1-SIB01 está configurado e iniciado.

### 2.3. Criar um Foreign Service Integration Bus

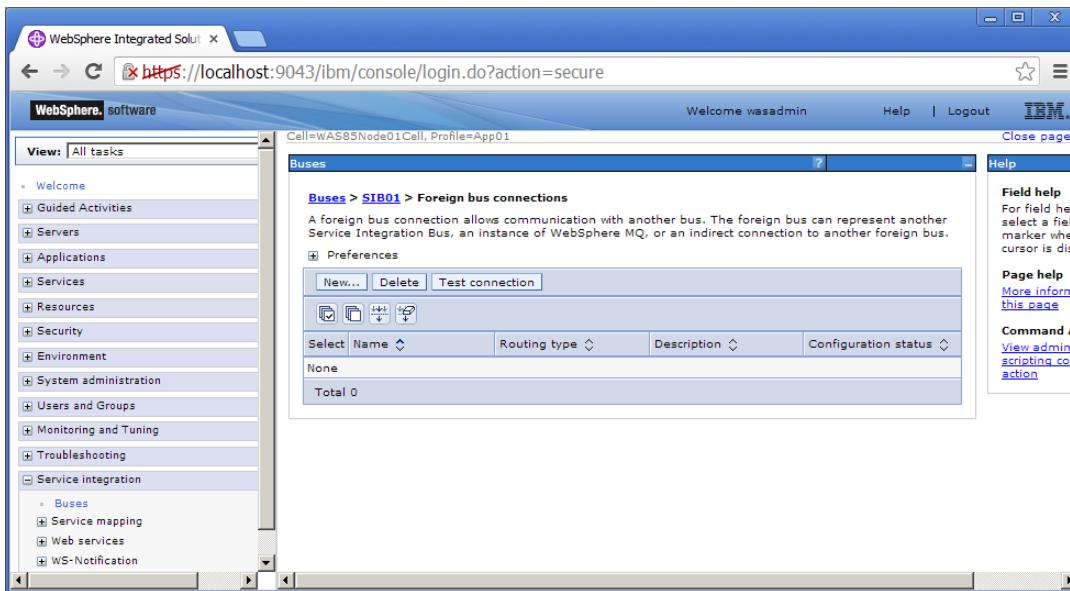
- Na console, expanda Service Integration e clique em Buses:



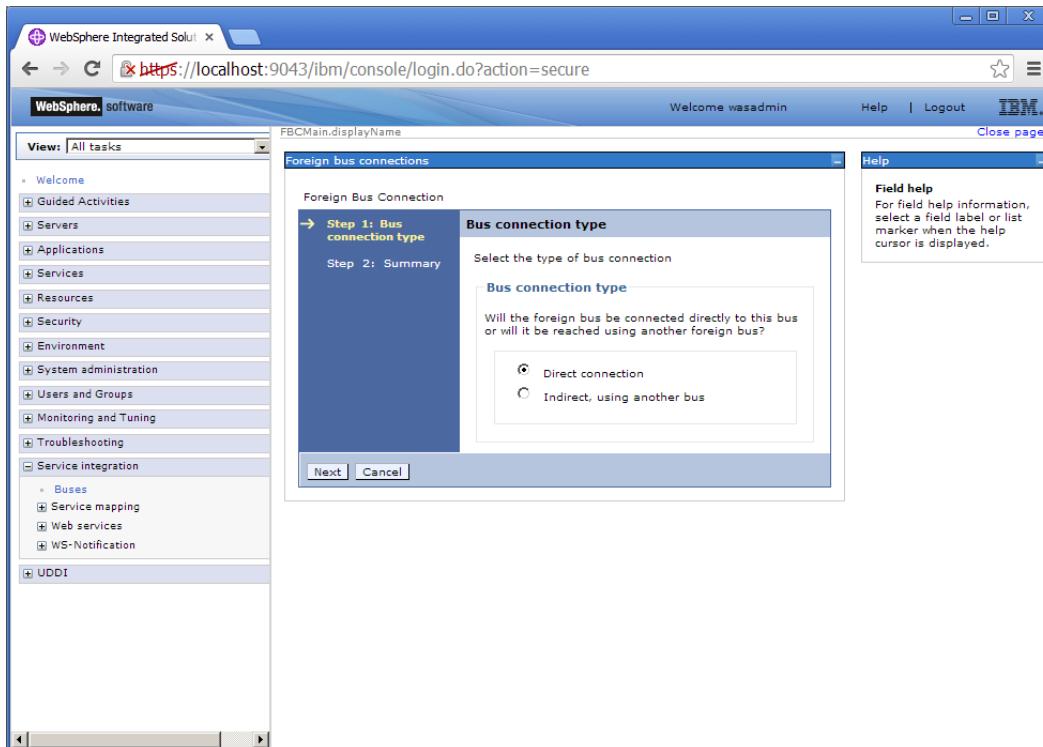
- Clique em SIB01:



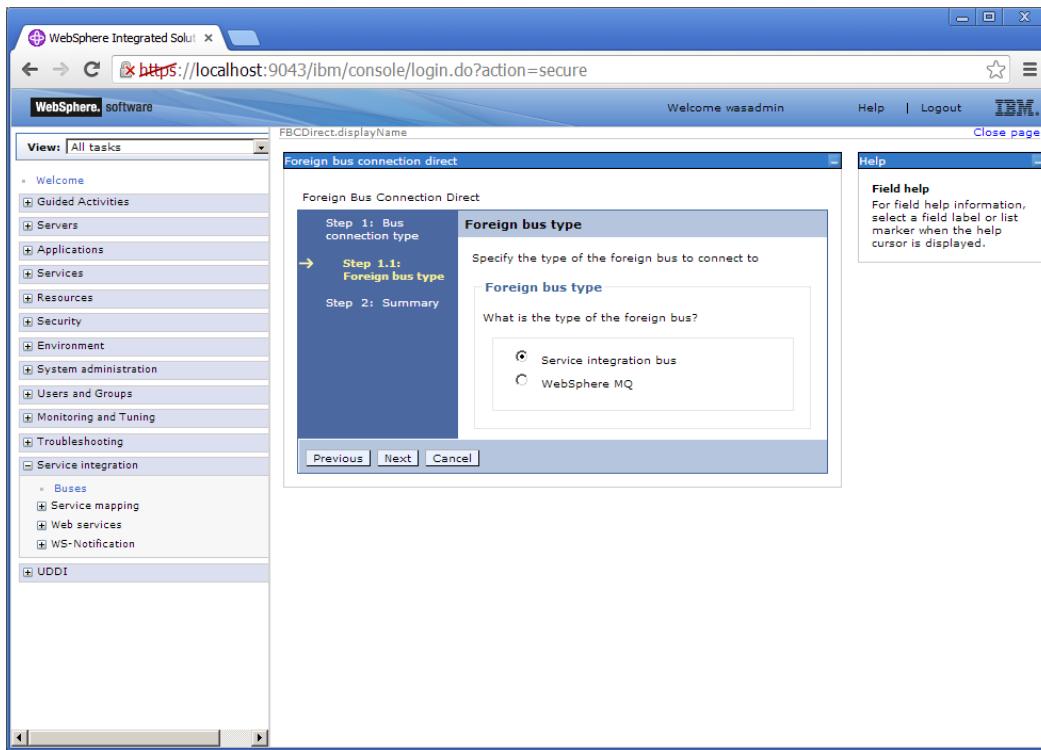
c) Clique em Topology-> Foreign bus connections:



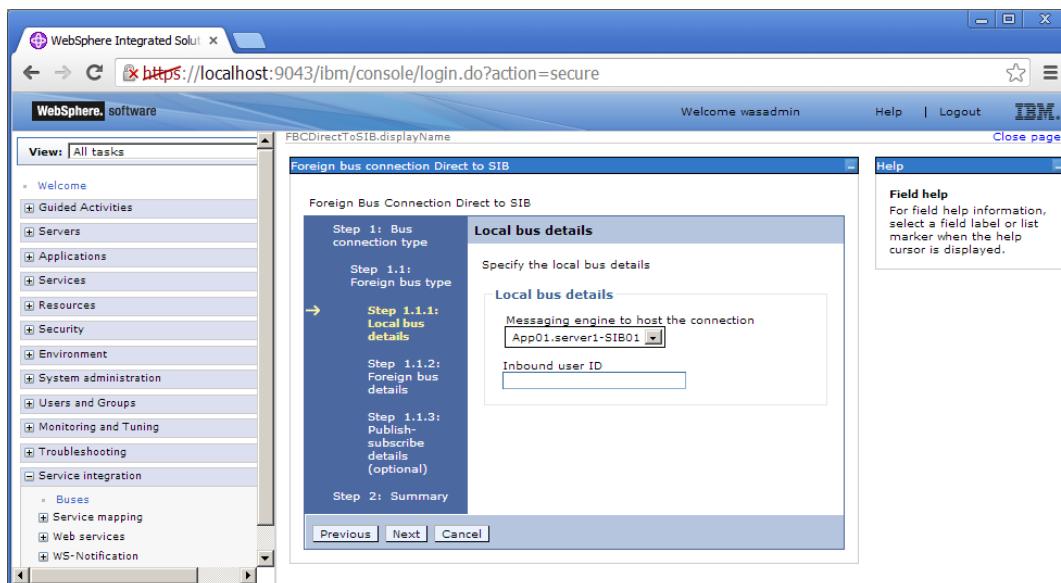
d) Clique em New:



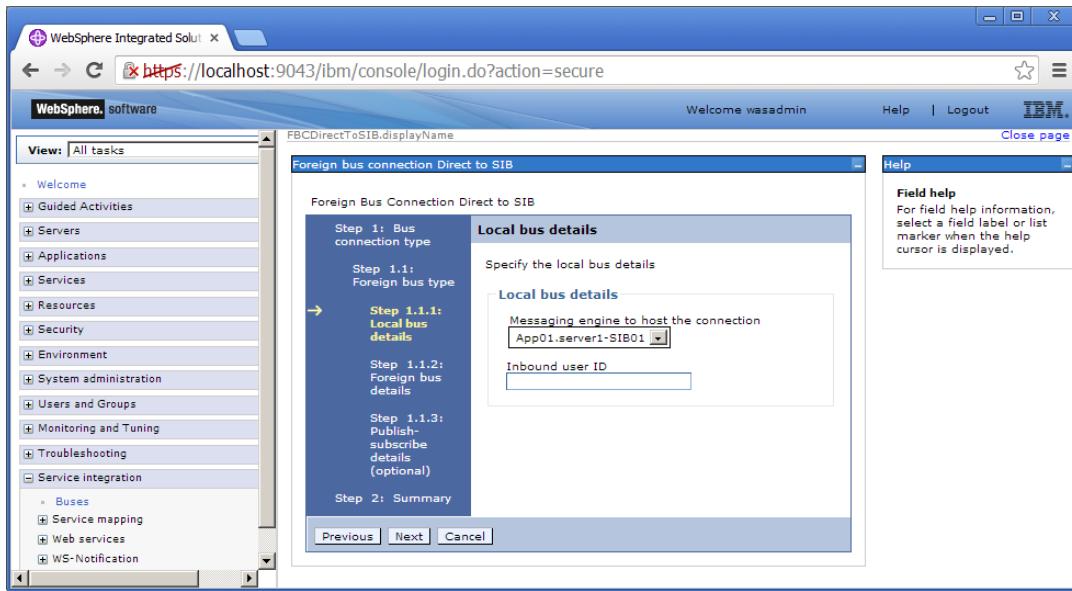
e) Escolha Direct connection e clique Next:



f) Escolha Service integration bus e clique Next:

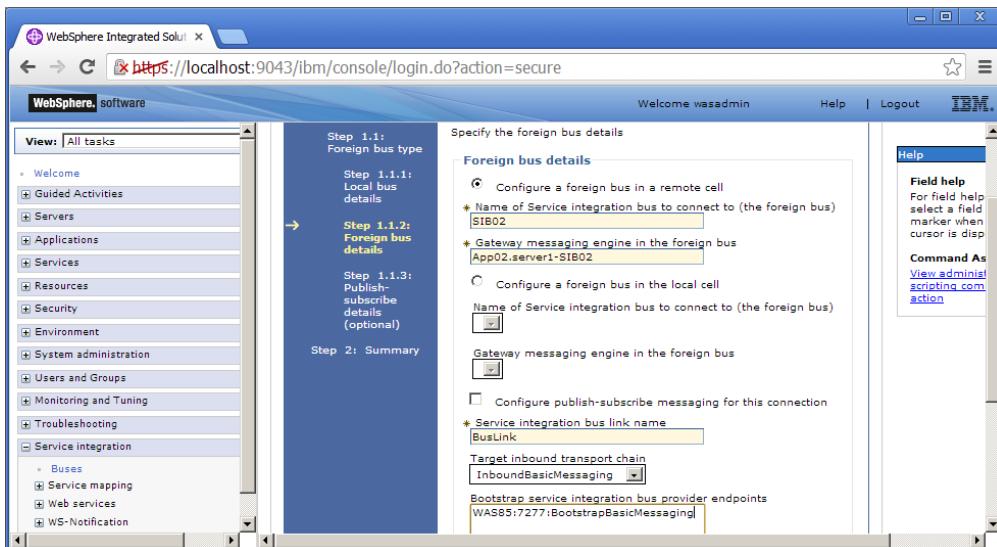


- g) Escolha o Messaging Engine App01.server1-SIB01 em Messaging engine to host the connection. Digite wasadmin em Inbound user ID e clique Next:

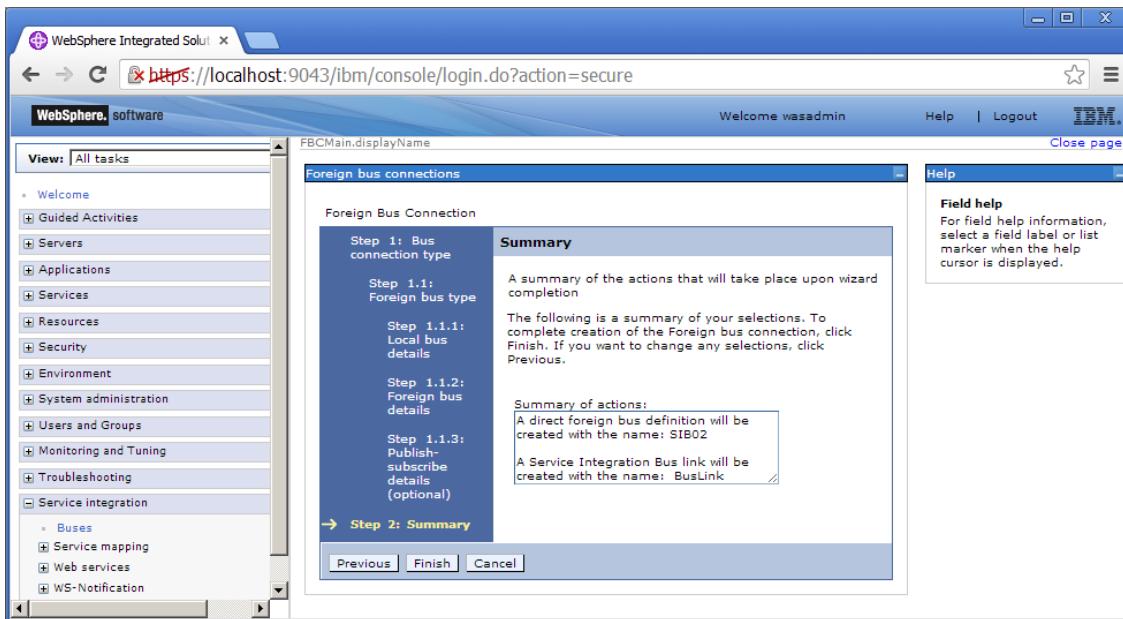


- h) Digite as seguintes configurações:

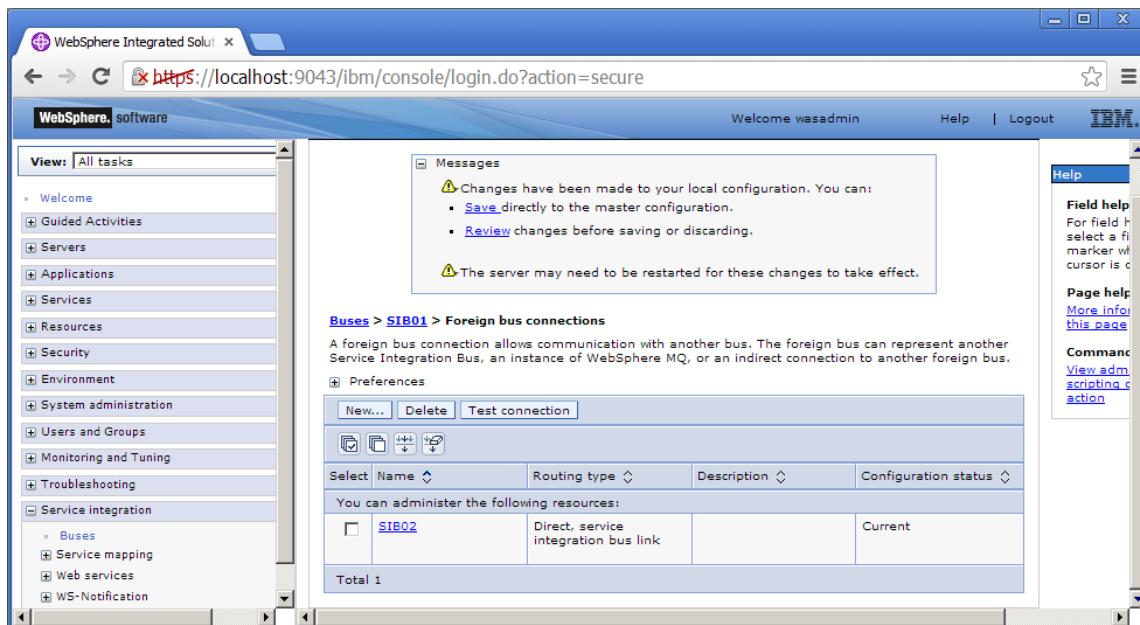
Name of Service integration bus to connect to (the foreign bus)	SIB02
Gateway messaging engine in the foreign bus	App02.server1-SIB02
Service integration bus link name	BusLink
Target inbound transport chain	InboundBasicMessaging
Bootstrap service integration bus provider endpoints	WAS85:7277:BootstrapBasicMessaging



i) Clique Next:



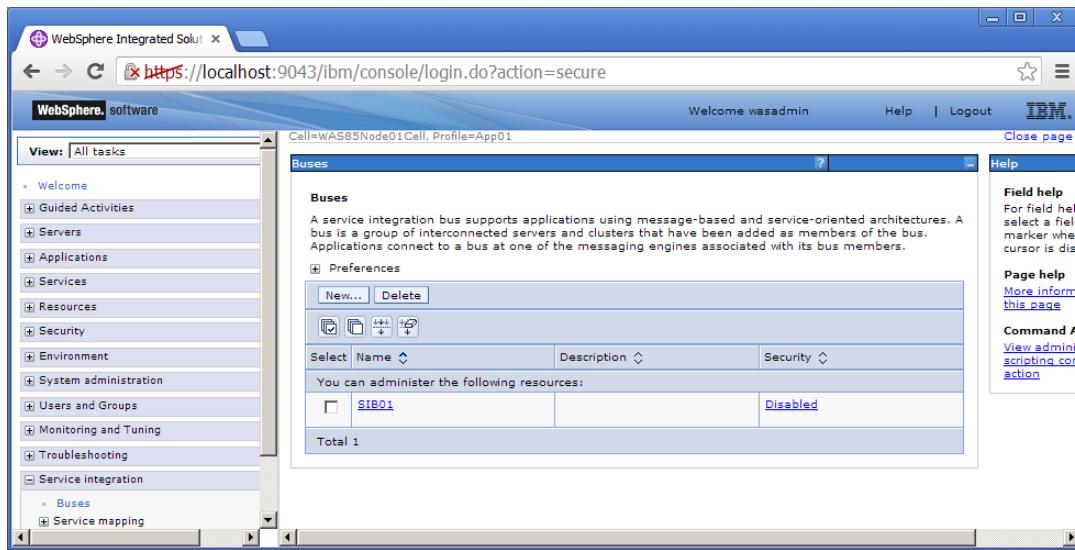
j) Reveja as configurações efetuadas e clique Finish:



k) Salve as configurações na Master Configuration.

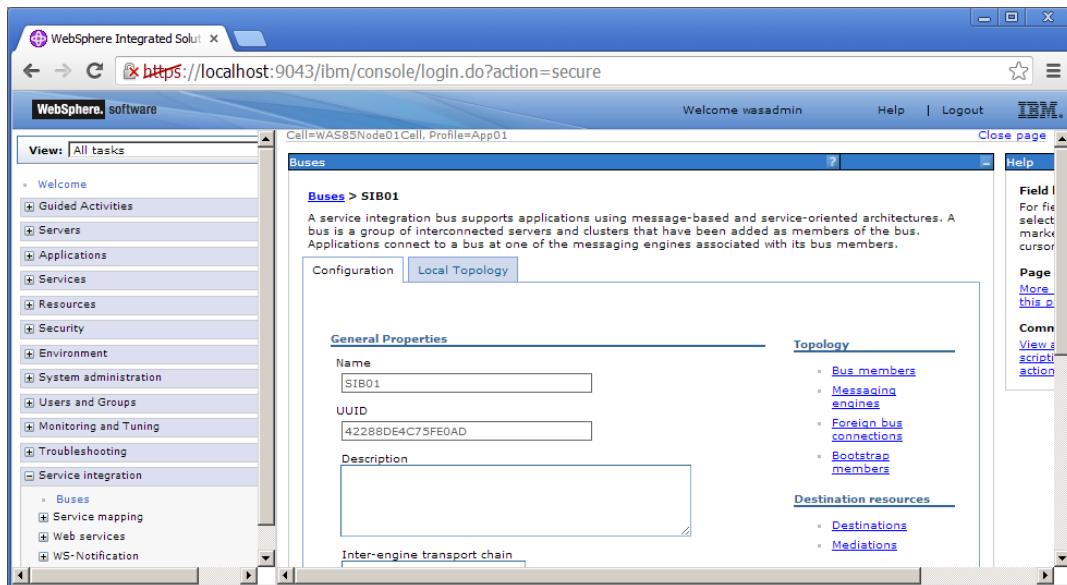
### 2.4. Criar uma Local Queue e uma Foreign Queue

- Na console, expanda Service Integration e clique em Buses:



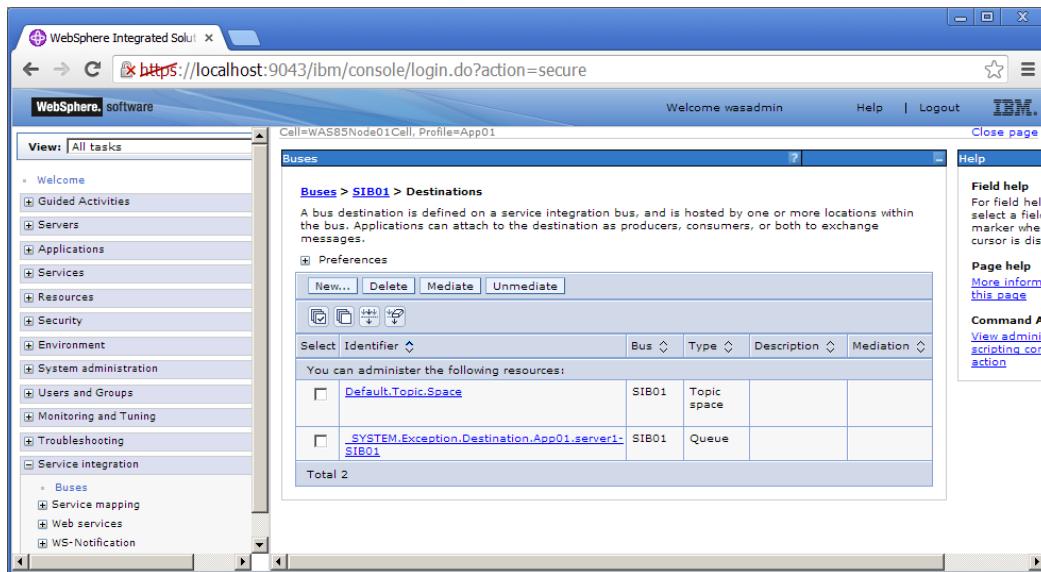
The screenshot shows the WebSphere Integrated Solutions Console interface. The left sidebar has a tree view with 'Service integration' expanded, showing 'Buses' and 'Service mapping'. The main panel is titled 'Buses' and contains a table with one row for 'SIB01'. The status column shows 'Disabled'.

- Clique em SIB01:

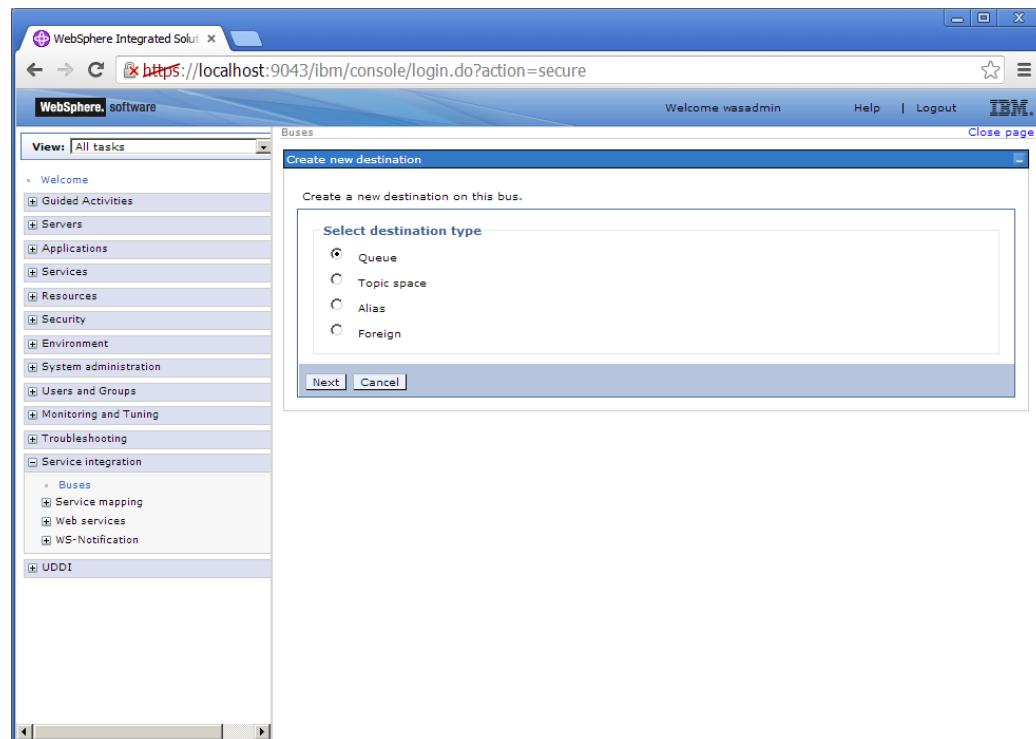


The screenshot shows the configuration details for the SIB01 bus. The 'General Properties' section includes fields for 'Name' (SIB01), 'UUID' (42288DE4C75FE0AD), and 'Description' (Inter-engine transport chain). The 'Topology' section lists 'Bus members', 'Messaging engines', 'Foreign bus connections', and 'Bootstrap members'. The 'Destination resources' section lists 'Destinations' and 'Mediations'.

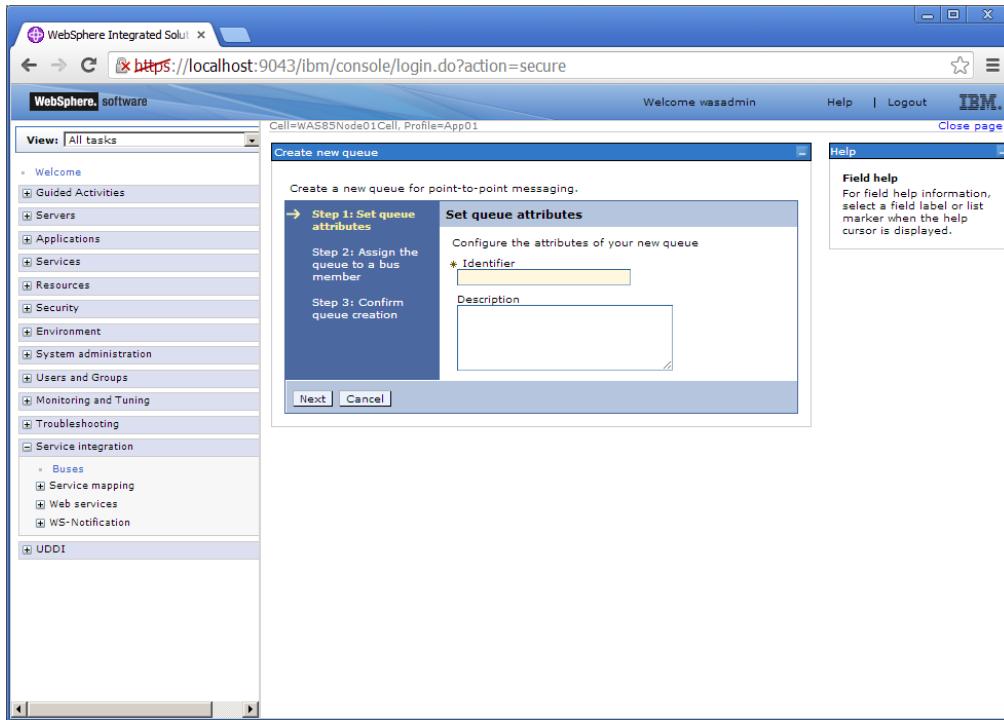
c) Clique em Destination Resources->Destinations:



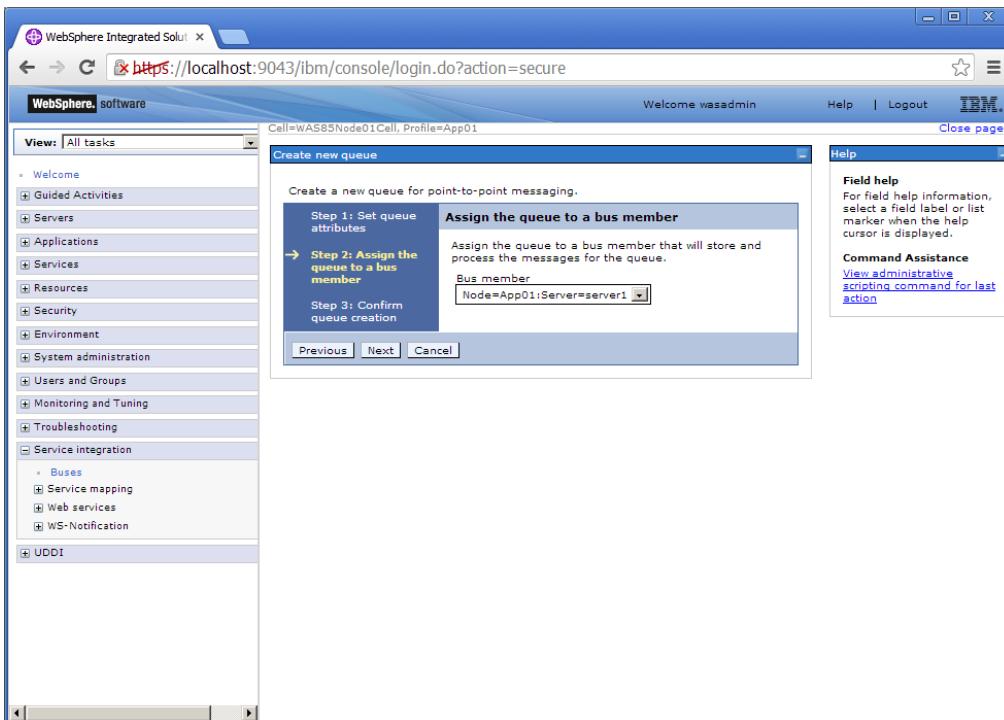
d) Clique New:



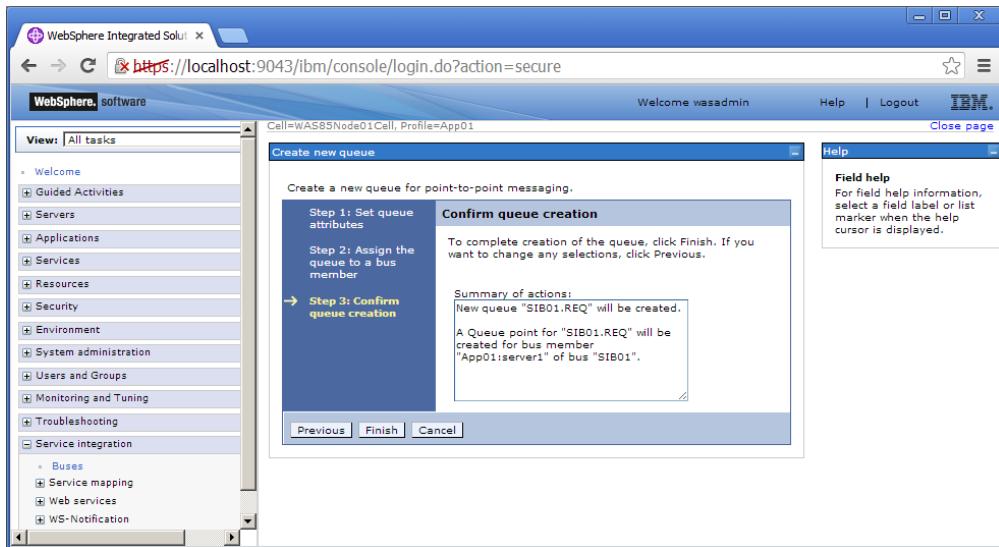
e) Selecione Queue e clique Next:



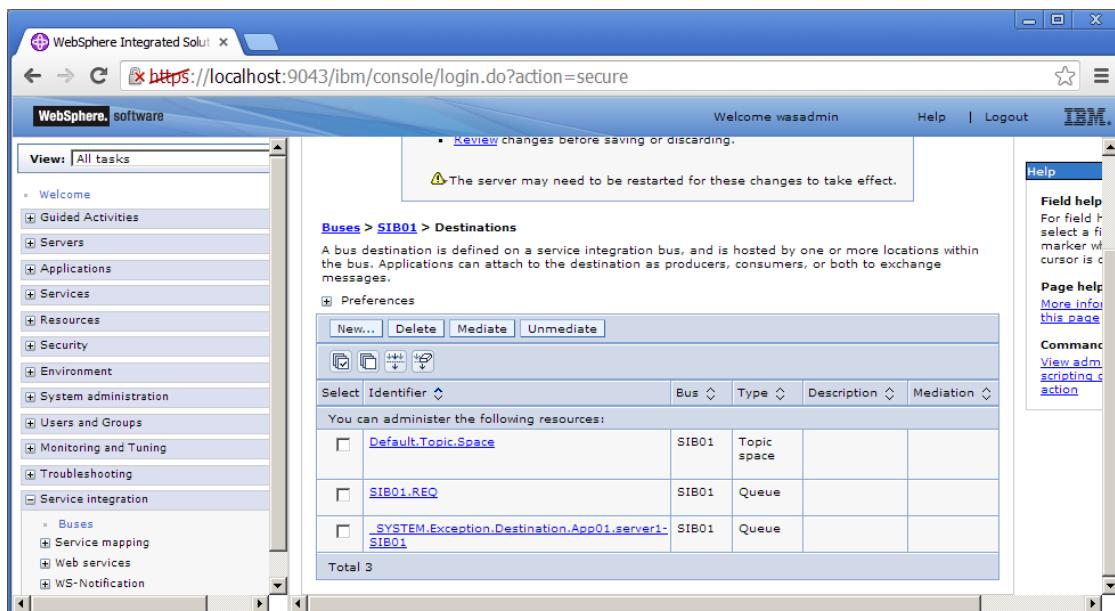
f) Digite SIB01.REQ em Identifier e clique Next:



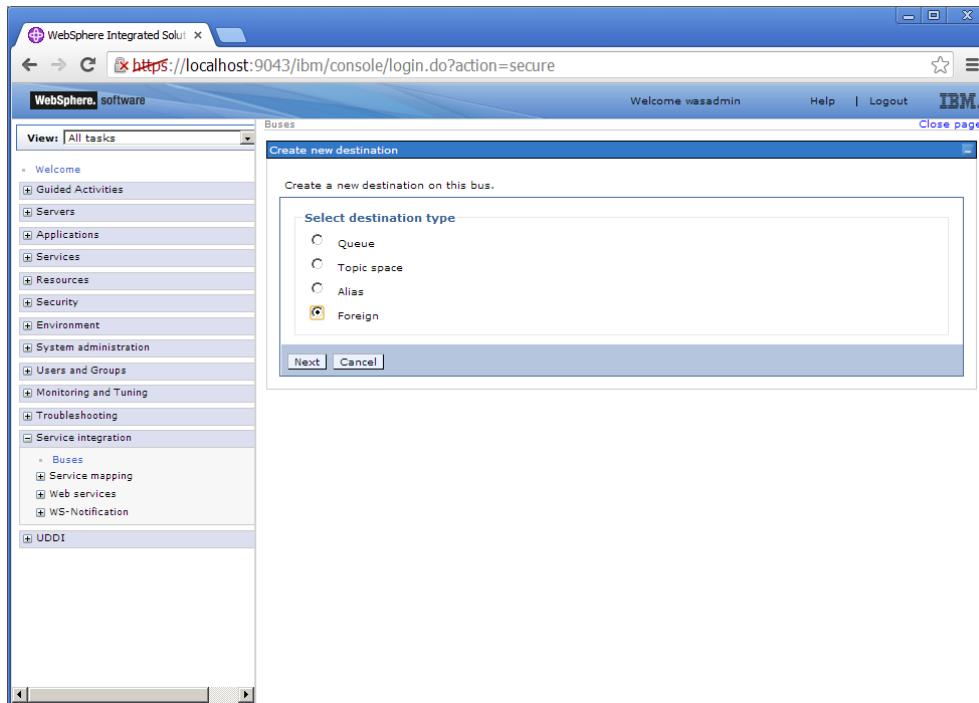
g) Selecione o Bus Member Node=App01:Server=server1 e clique Next:



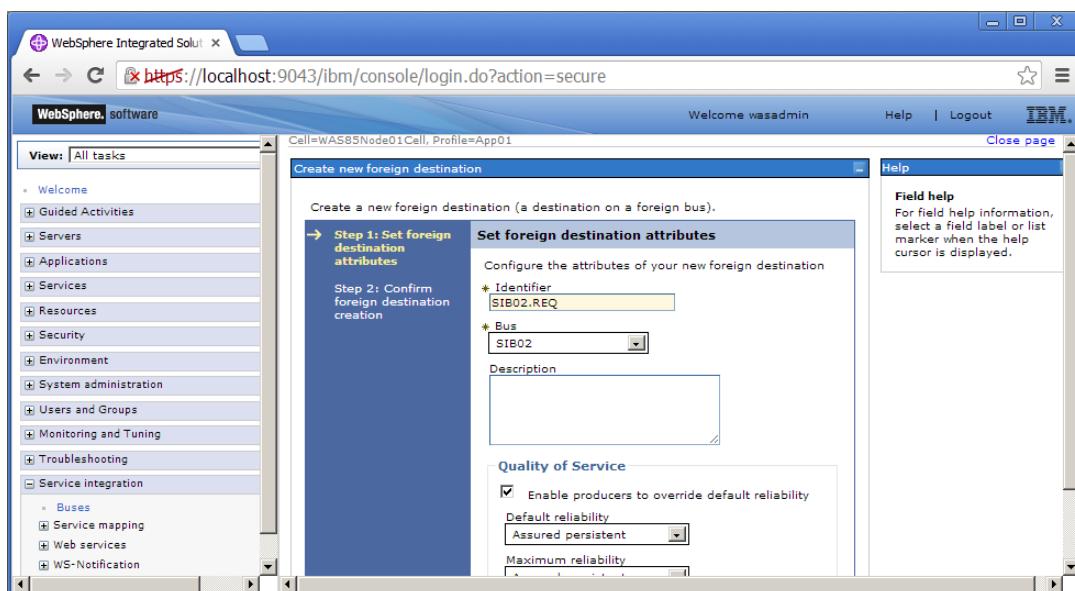
h) Revise as configurações e clique Finish:



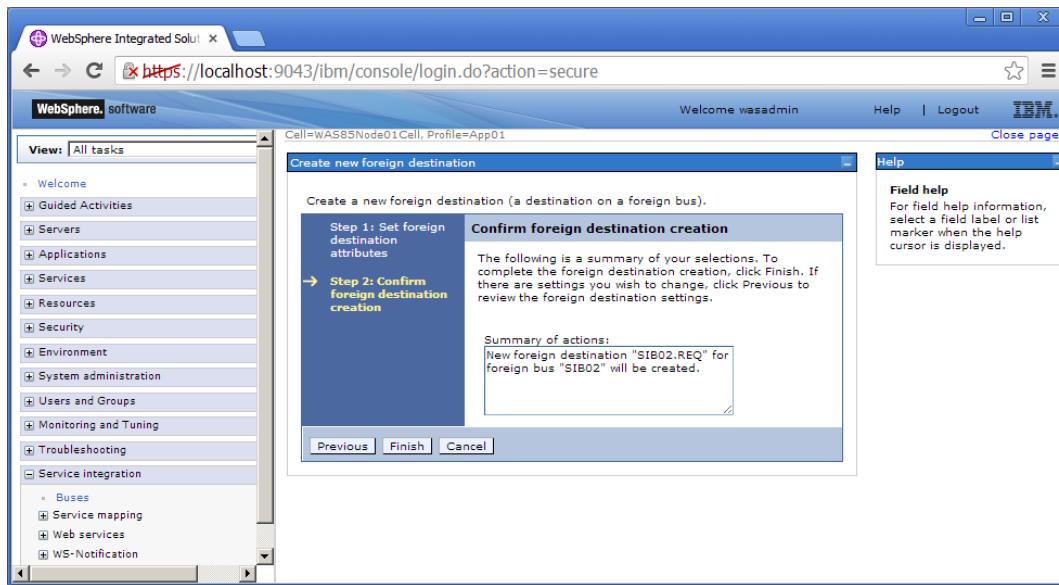
- i) A fila local está configurada.
- j) Clique em New:



- k) Selecione Foreign e clique Next:



- I) Digite SIB02.REQ em Identifier, selecione SIB02 em Bus e clique Next:



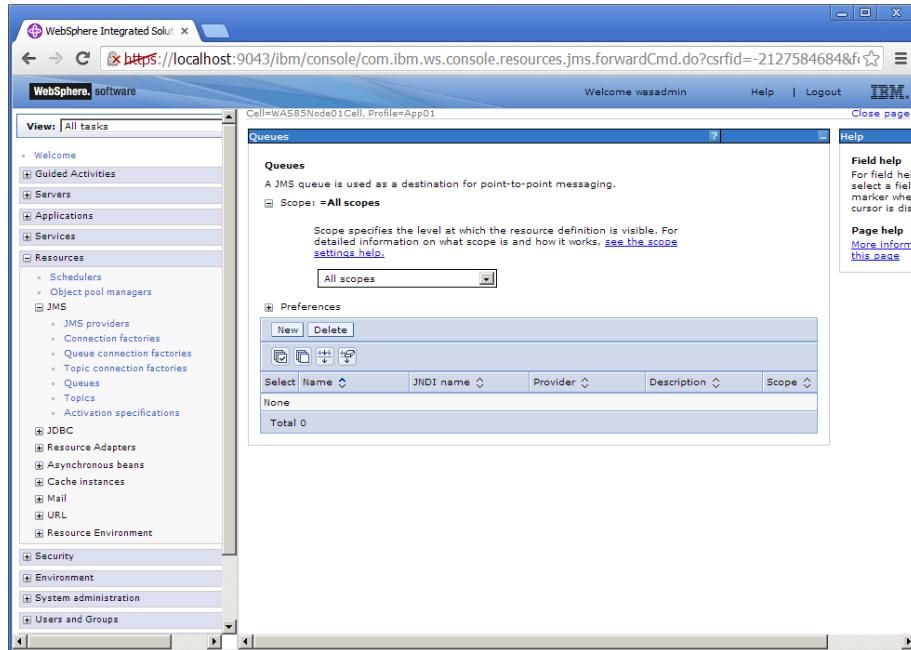
- m) Revise as configurações efetuadas e clique Finish:

Select	Identifier	Bus	Type	Description	Mediation
<input type="checkbox"/>	<a href="#">Default.Topic.Space</a>	SIB01	Topic space		
<input type="checkbox"/>	<a href="#">SIB01.REQ</a>	SIB01	Queue		
<input type="checkbox"/>	<a href="#">SIB02.REQ</a>	SIB02	Foreign		
<input type="checkbox"/>	<a href="#">SYSTEM.Exception.Destination.App01.server1-SIB01</a>	SIB01	Queue		

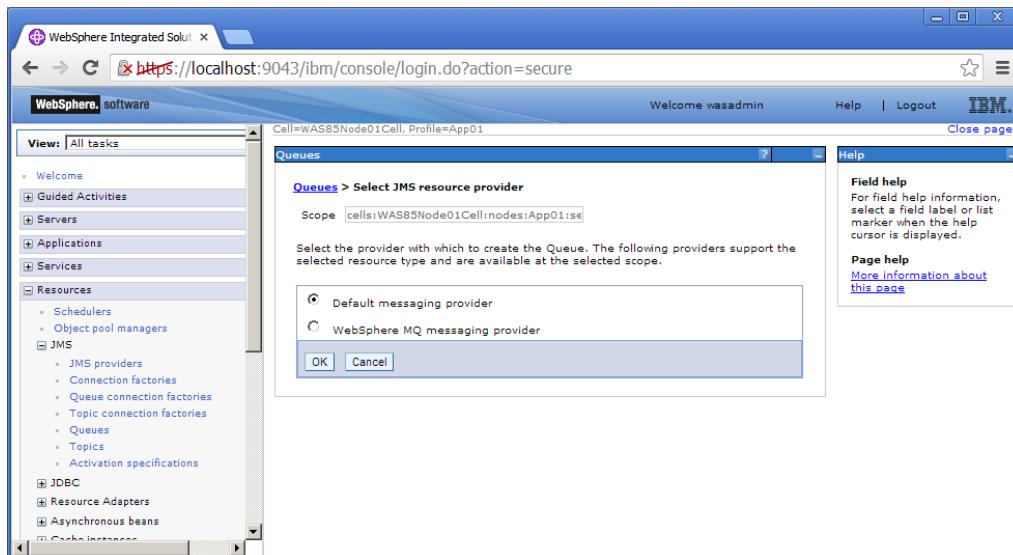
- n) Salve as configurações na Master Configuration.

### 2.5. Criar filas JMS

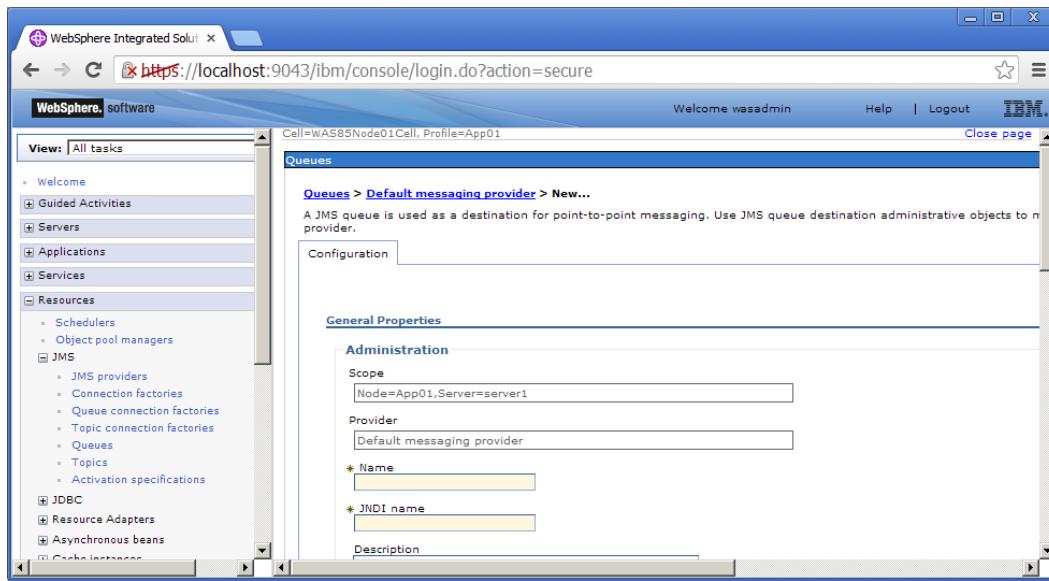
- a) Na console administrativa, expanda Resources->JMS e clique em Queues:



- b) Selecione o escopo Node=App01, Server=server1 e clique em New:

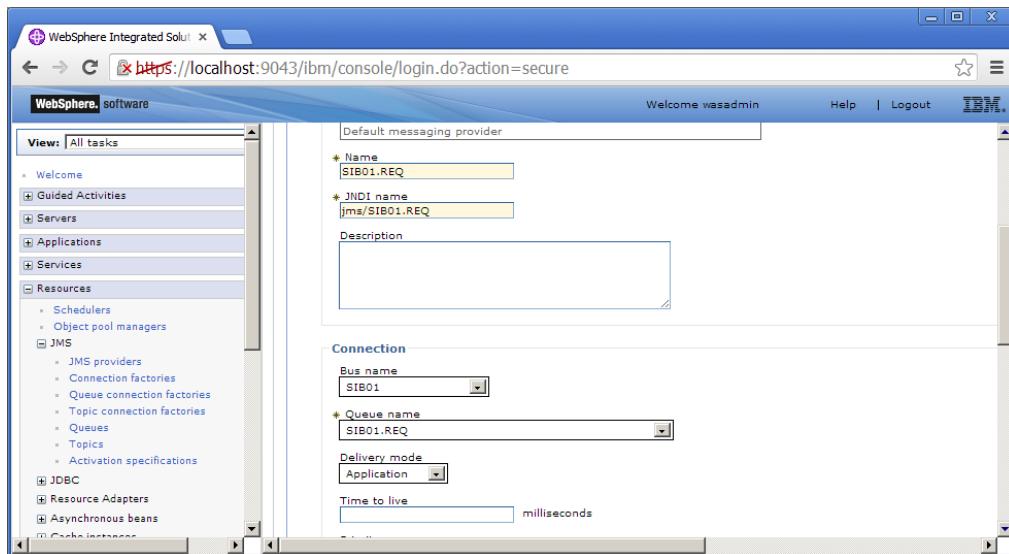


c) Escolha Default Messaging Provider e clique OK:

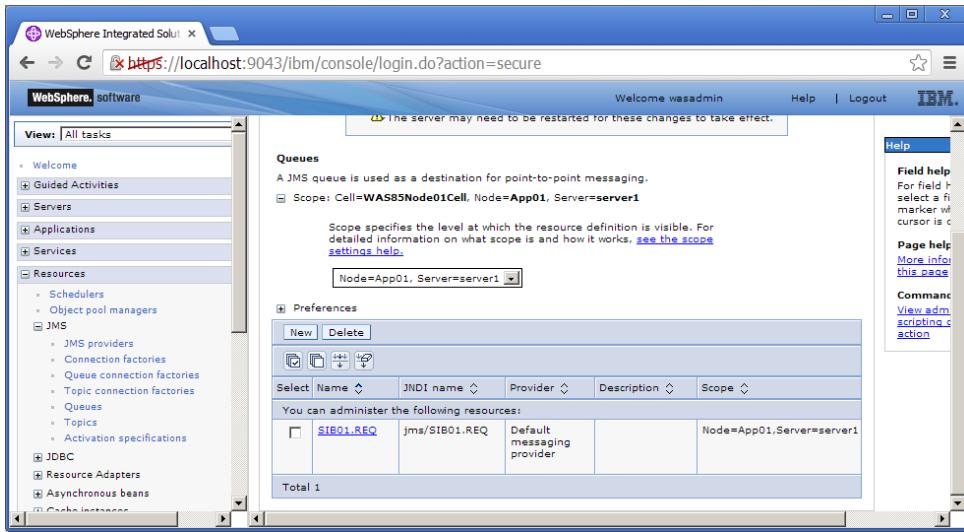


d) Digite as seguintes informações:

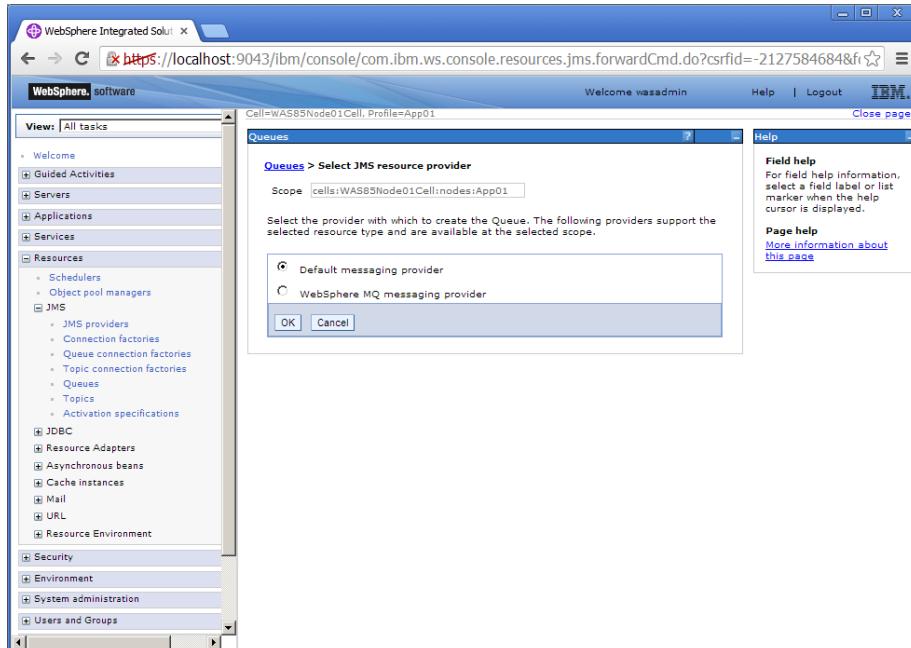
Name	SIB01.REQ
JNDI Name	jms/SIB01.REQ
Bus name	SIB01
Queue name	SIB01.REQ



e) Clique OK:

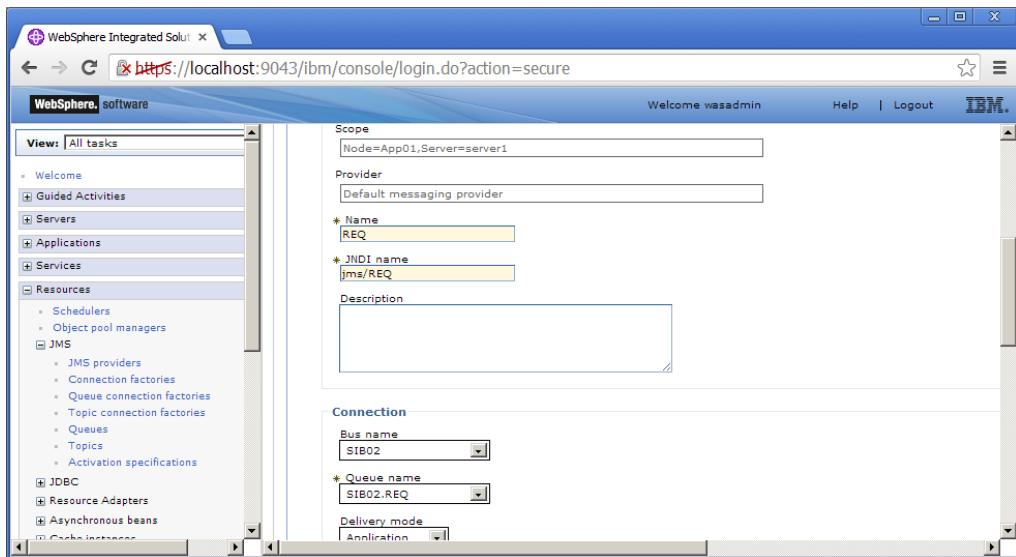


- f) Salve as configurações na Master Configuration.  
g) Clique em New:

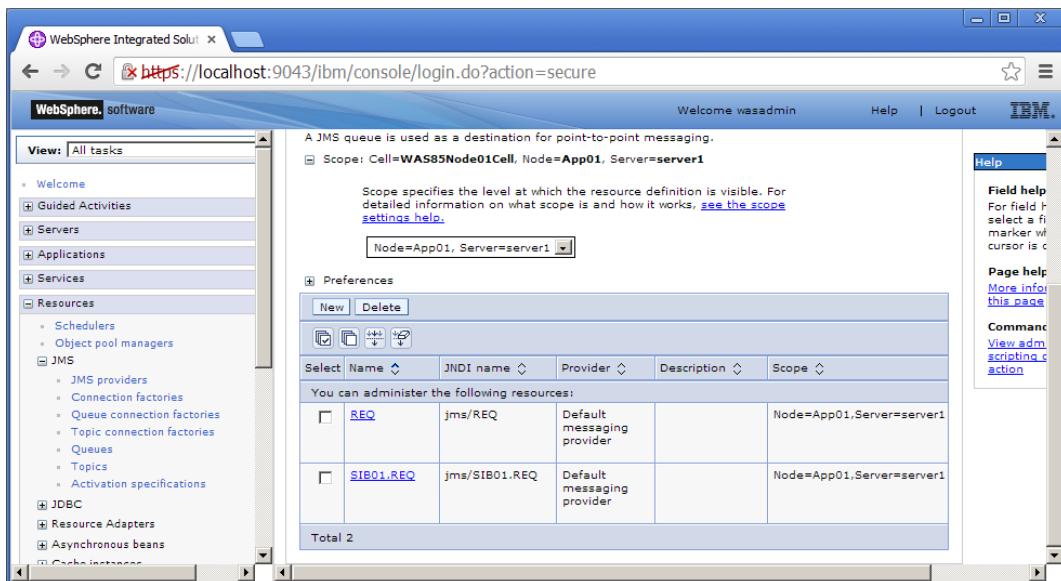


- h) Selecione Default messaging provider e clique OK:  
 i) Digite as seguintes informações:

Name	REQ
JNDI Name	jms/REQ
Bus name	SIB02
Queue name	SIB02.REQ



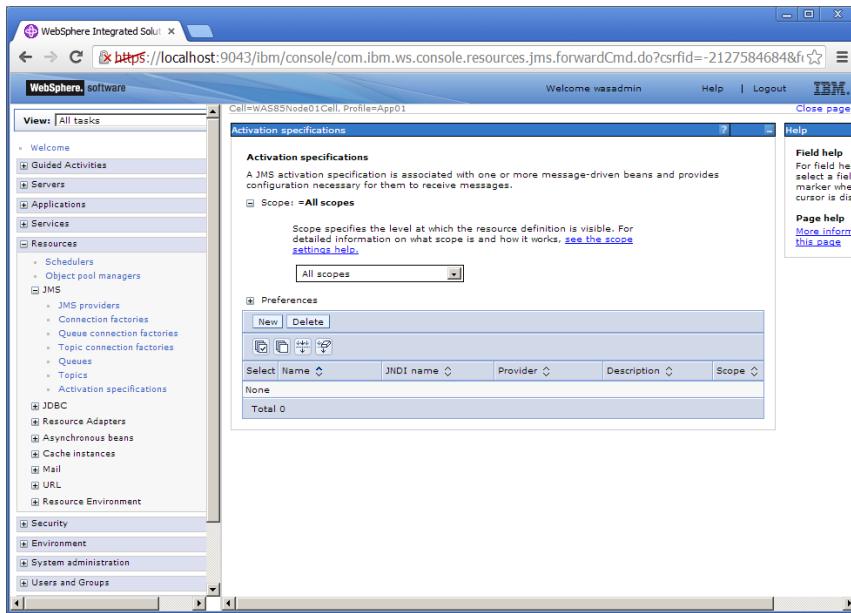
- j) Clique OK:



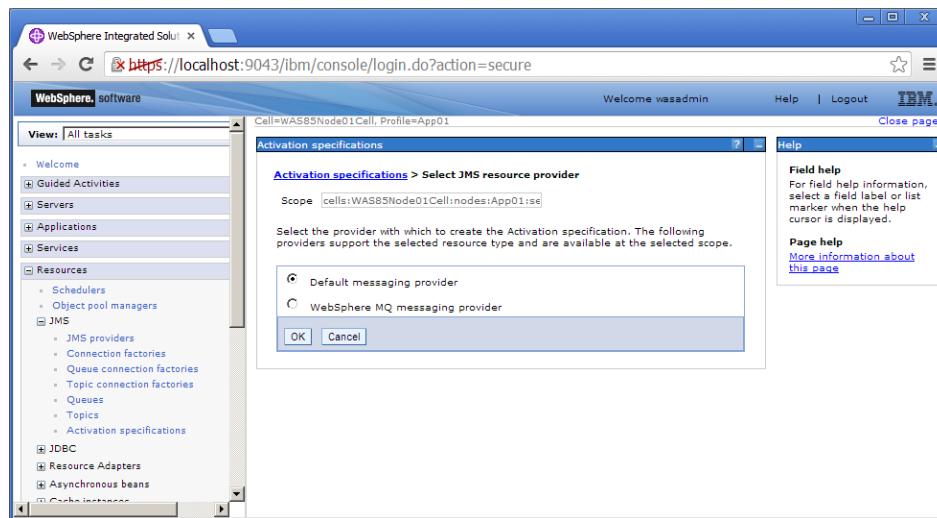
- k) Salve as configurações na Master Configuration.

### 2.6. Criar um Activation Spec

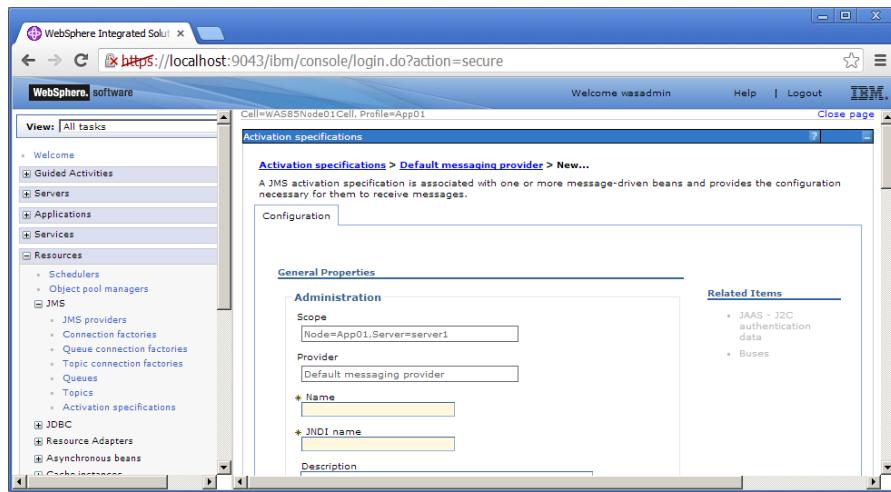
- a) Na console administrativa, expanda Resources->JMS e clique em Activation specifications:



- b) Selecione o escopo Node=App01, Server=server1 e clique em New:

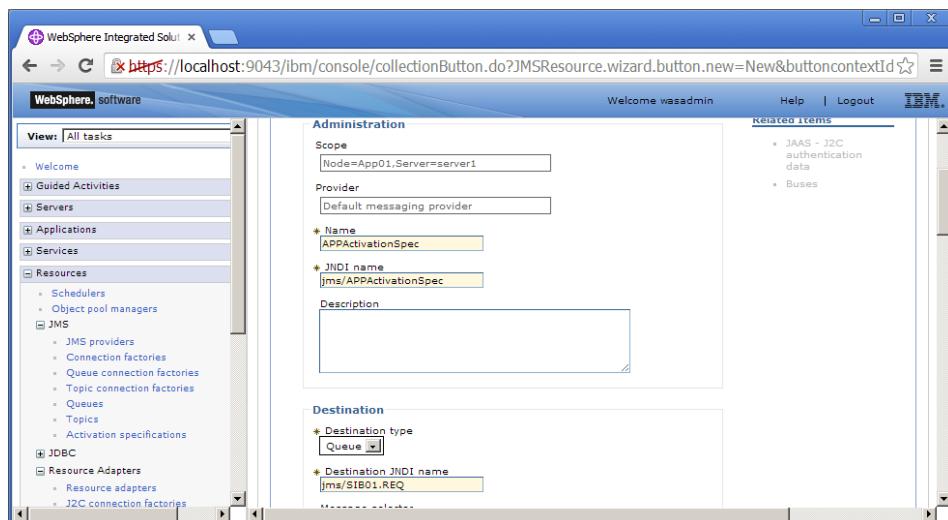


c) Escolha Default Messaging Provider e clique OK:

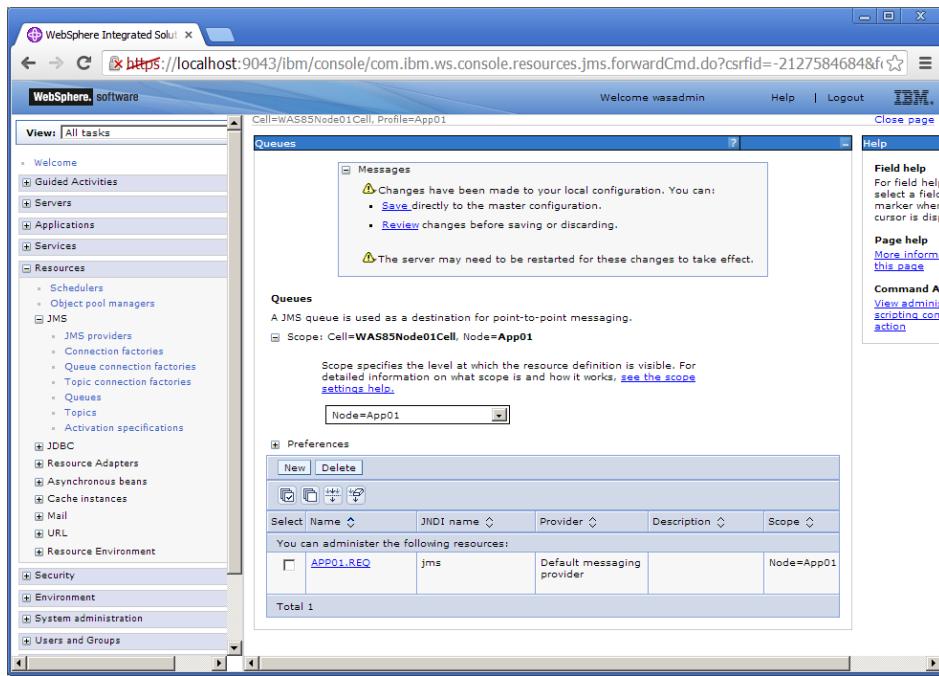


d) Digite as seguintes informações:

Name	APPActivationSpec
JNDI Name	jms/APPActivationSpec
Destination Type	Queue
Destination JNDI Type	jms/SIB01.REQ
Bus name	SIB01
Target inbound transport chain	InboundBasicMessaging
Provider endpoints	WAS85:7276:BootstrapBasicMessaging



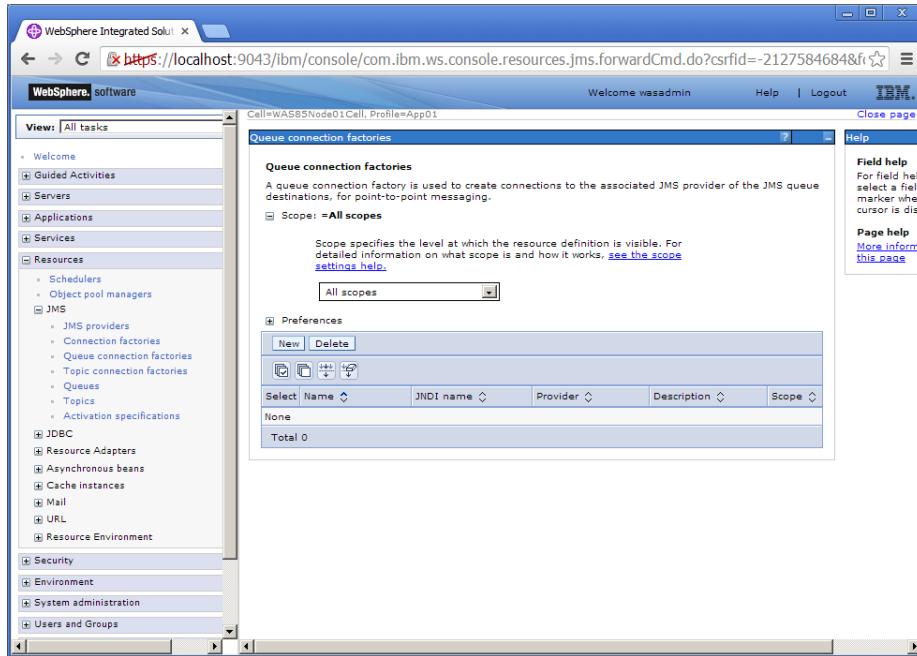
e) Clique OK:



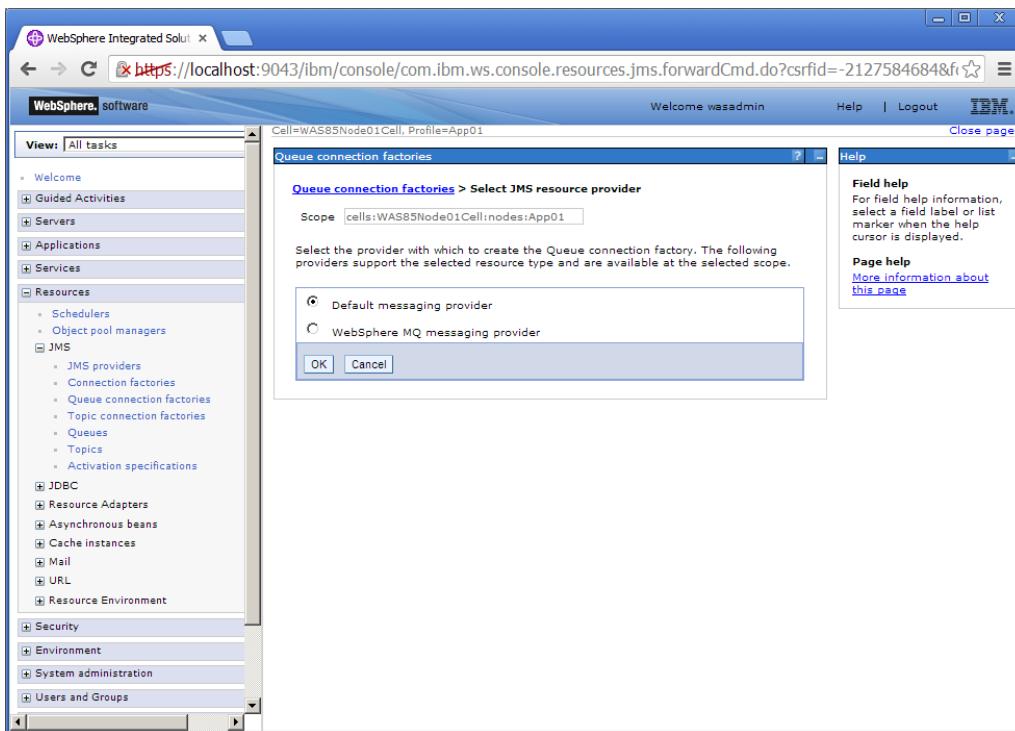
f) Salve as configurações na Master Configuration.

### 2.7. Criar um Queue Connection Factory

- a) Na console administrativa, expanda Resources->JMS e clique em Queue Connection Factories:

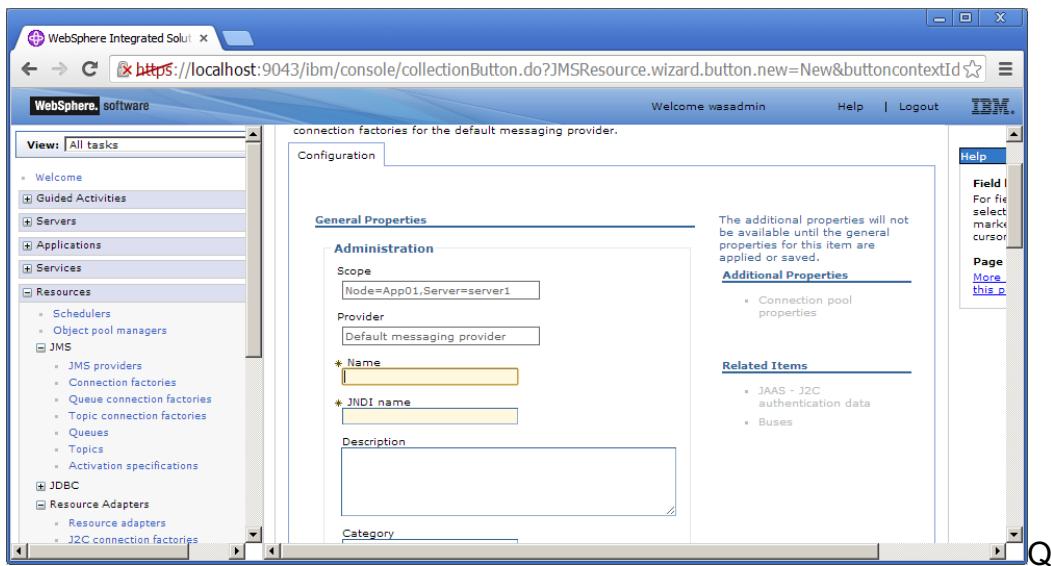


- b) Selecione o escopo Node=App01, Server=server1 e clique em New:



- c) Escolha Default Messaging Provider e clique OK:

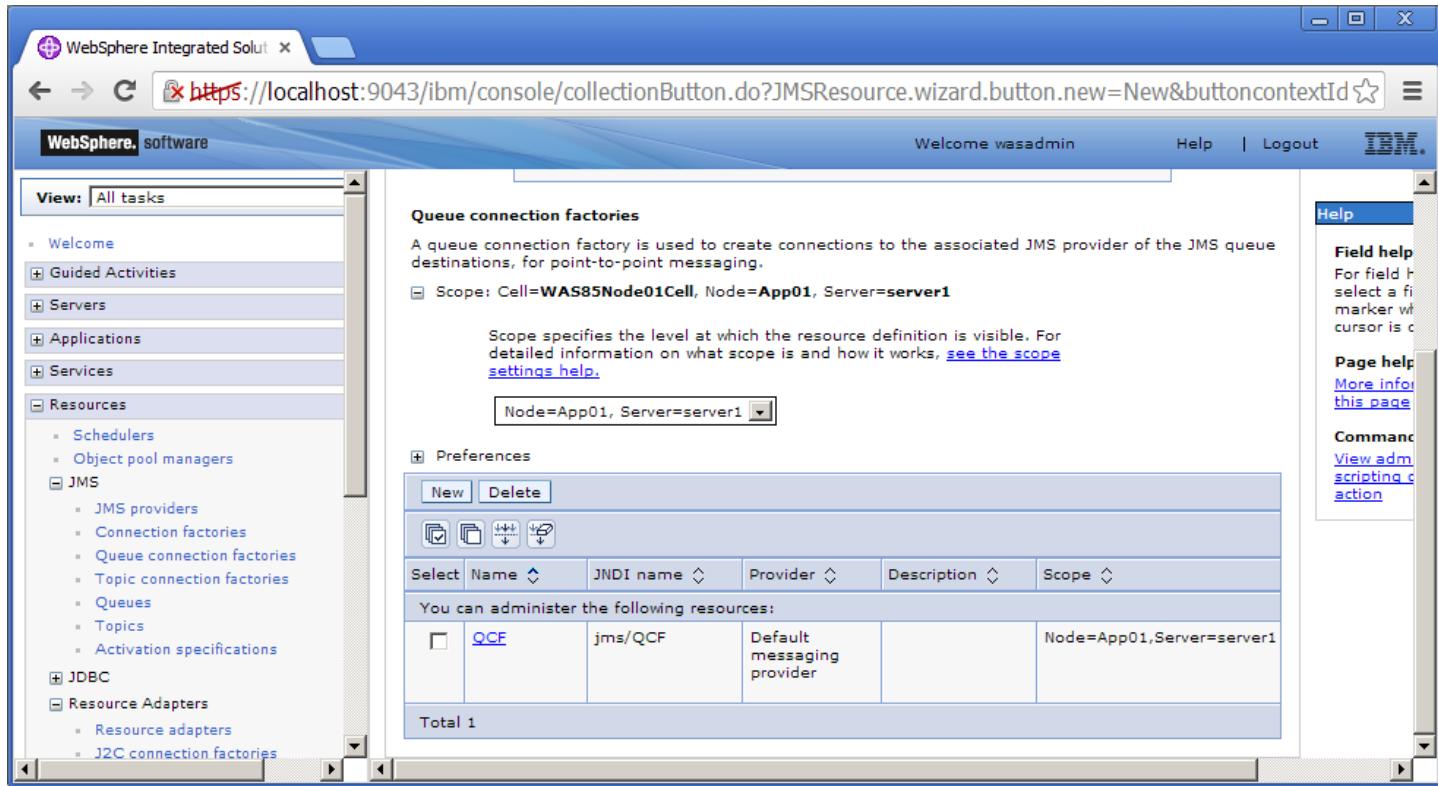
As informações apresentadas neste documento são de propriedade da Certsys, e seu conteúdo pode ser utilizado internamente pelo cliente citado, para avaliação de seus termos, aprovação e acompanhamento do projeto. **Não pode ser copiado, totalmente ou em partes, sem a autorização expressa da Certsys.**



d) Digite as seguintes informações:

Name	QCF
JNDI Name	jms/QCF
Bus name	SIB01
Target inbound transport chain	InboundBasicMessaging
Provider endpoints	WAS85:7276:BootstrapBasicMessaging

e) Clique OK:



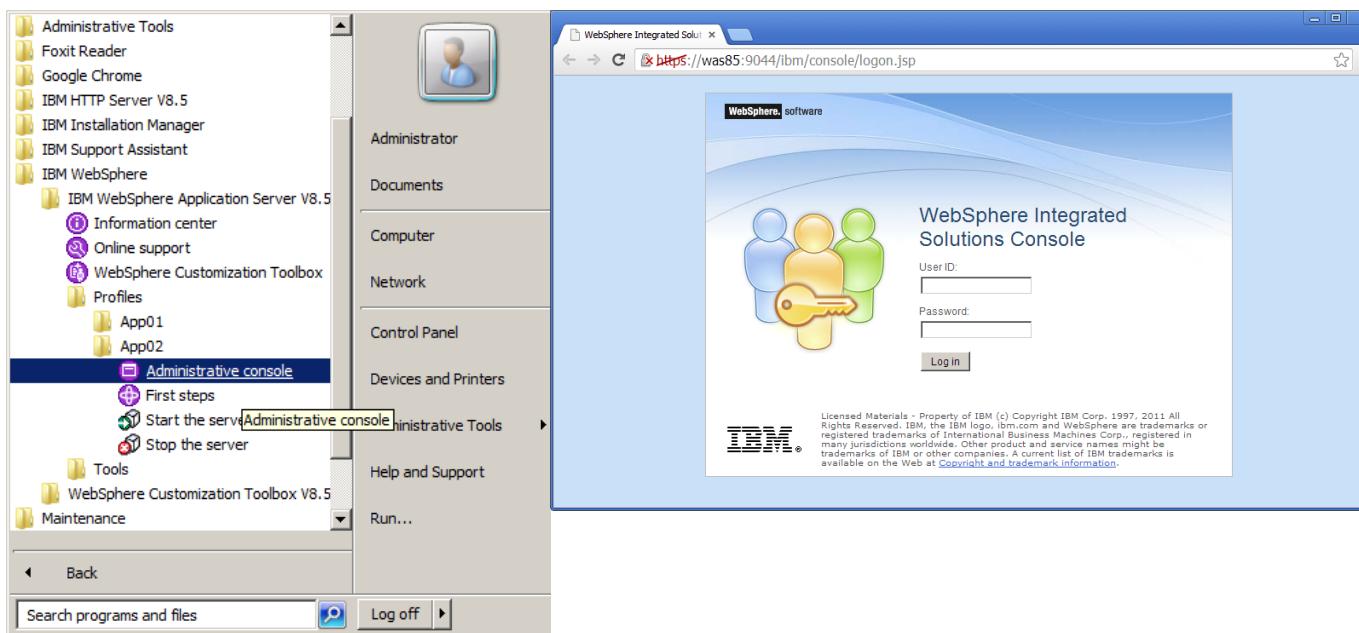
Select	Name	JNDI name	Provider	Description	Scope
<input type="checkbox"/>	<a href="#">QCF</a>	jms/QCF	Default messaging provider		Node=App01,Server=server1

f) Salve as configurações na Master Configuration.

### 3. Configuração do Nó App02

#### 3.1. Configuração do Service Integration Bus SIB02

- Abra os serviços do Windows e verifique se os serviços referentes aos servidores de aplicação estão no ar:
- Abra a console administrativa do App02 (<http://localhost:9061/admin>) ou pelo atalho:



- Digite o usuário wasadmin / Cert0104sys e clique Login. No menu a esquerda, expanda Service Integration e clique em Buses:
- Clique New. Digite SIB02 em Enter the name for your new bus e DESMARQUE a opção Bus security:
- Clique Next. Reveja as configurações feitas:
- Clique Finish para criar o Service Integration Bus SIB02:
- Salve as configurações na Master Configuration.

### 3.2. Criar um Bus Member para o Service Integration Bus

- a) Na console administrativa, expanda Service Integration e clique em Buses:
- b) Clique em SIB02:
- c) Clique em Topology-> Bus Members:
- d) Clique em Add para adicionar um Bus Member ao Service Integration Bus:
- e) Selecione Server (App02:server1) e clique Next:
- f) Selecione File Store e Clique Next:
- g) Mantenha as configurações padrões e clique Next:
- h) Mantenha as configurações padrões e clique Next:
- i) Revise as configurações realizadas e clique Finish:
- j) Salve as configurações feitas na Master Configuration. Faça um logout da console.
- k) Nos serviços do Windows, reinicie o serviço IBM WebSphere Application Server V8.5 - App02:
- l) Entre novamente na Console Administrativa.
- m) Expand Service Integration e Clique em Buses.
- n) Clique em SIB02.
- o) Clique em Topology->Messaging Engines:
- p) Verifique que o Messaging Engine App02.server1-SIB02 está configurado e iniciado.

### 3.3. Criar um Foreign Service Integration Bus

- a) Na console, expanda Service Integration e clique em Buses:
- b) Clique em SIB02:
- c) Clique em Topology-> Foreign bus connections:
- d) Clique em New:
- e) Escolha Direct connection e clique Next:
- f) Escolha Service integration bus e clique Next:
- g) Escolha o Messaging Engine App02.server1-SIB02 em Messaging engine to host the connection. Digite wasadmin em Inbound user ID e clique Next:
- h) Digite as seguintes configurações:

Name of Service integration bus to connect to (the foreign bus)	SIB01
Gateway messaging engine in the foreign bus	App01.server1-SIB01
Service integration bus link name	BusLink
Target inbound transport chain	InboundBasicMessaging
Bootstrap service integration bus provider endpoints	WAS85:7276:BootstrapBasicMessaging

- i) Clique Finish:
- j) Salve as configurações na Master Configuration.

### 3.4. Criar uma Local Queue e uma Foreign Queue

- a) Na console, expanda Service Integration e clique em Buses:
- b) Clique em SIB02:
- c) Clique em Destination Resources->Destinations:
- d) Clique New:
- e) Selecione Queue e clique Next:
- f) Digite SIB02.REQ em Identifier e clique Next:
- g) Selecione o Bus Member Node=App02:Server=server1 e clique Next:
- h) Revise as configurações e clique Finish:
- i) A fila local está configurada.
- j) Clique em New:
- k) Selecione Foreign e clique Next:
- l) Digite SIB01.REQ em Identifier, selecione SIB01 em Bus e clique Next:
- m) Revise as configurações efetuadas e clique Finish:
- n) Salve as configurações na Master Configuration.

### 3.5. Criar filas JMS

- a) Na console administrativa, expanda Resources->JMS e clique em Queues:
- b) Selecione o escopo Node=App02, Server=server1 e clique em New:
- c) Escolha Default Messaging Provider e clique OK:
- d) Digite as seguintes informações:

Name	SIB02.REQ
JNDI Name	jms/SIB02.REQ
Bus name	SIB02
Queue name	SIB02.REQ

- e) Clique OK:
- f) Salve as configurações na Master Configuration.
- g) Clique em New:
- h) Selecione Default messaging provider e clique OK:
- i) Digite as seguintes informações:

Name	REQ
JNDI Name	jms/REQ
Bus name	SIB01
Queue name	SIB01.REQ

- j) Clique OK:
- k) Salve as configurações na Master Configuration.



### 3.6. Criar um Activation Spec

- a) Na console administrativa, expanda Resources->JMS e clique em Activation specifications:
- b) Selecione o escopo Node=App02, Server=server1 e clique em New:
- c) Escolha Default Messaging Provider e clique OK:
- d) Digite as seguintes informações:

Name	APPActivationSpec
JNDI Name	jms/APPActivationSpec
Destination Type	Queue
Destination JNDI Type	jms/SIB02.REQ
Bus name	SIB02
Target inbound transport chain	InboundBasicMessaging
Provider endpoints	WAS85:7277:BootstrapBasicMessaging

- e) Clique OK:
- f) Salve as configurações na Master Configuration.

### 3.7. Criar um Queue Connection Factory

- a) Na console administrativa, expanda Resources->JMS e clique em Queue Connection Factories:
- b) Selecione o escopo Node=App02, Server=server1 e clique em New:
- c) Escolha Default Messaging Provider e clique OK:
- d) Digite as seguintes informações:

Name	QCF
JNDI Name	jms/QCF
Bus name	SIB02
Target inbound transport chain	InboundBasicMessaging
Provider endpoints	WAS85:7277:BootstrapBasicMessaging

- e) Clique OK:
- f) Salve as configurações na Master Configuration.

### 4. Desenvolvimento da aplicação de teste da mensageria

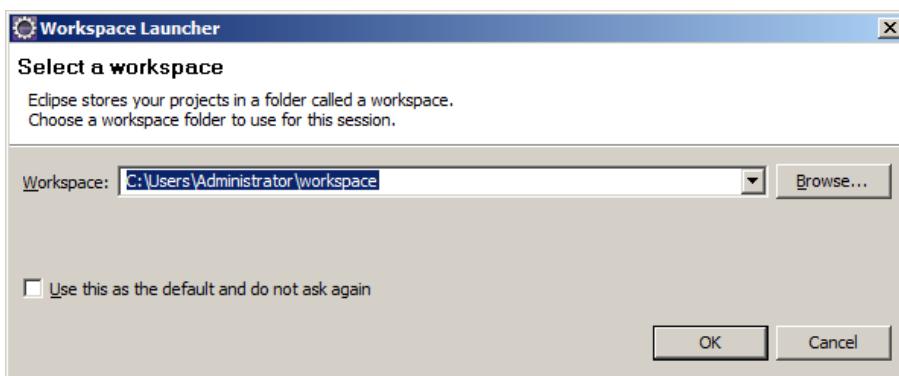
A aplicação de teste de mensageria é composta por um Message Driven Bean que lê da fila local do Service Integration Bus e de um Session Bean, que escreve na fila remota do SIB. O Session Bean está exposto como um Web Service, logo utilizaremos o SOAPUI para testar a chamada a essa requisição.

O fluxo da informação a ser validado é o seguinte:

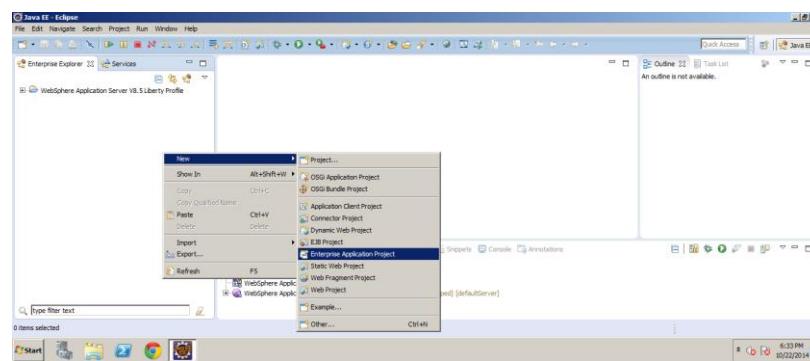
Mensagem no SOAPUI -> WebService App01 -> Session Bean App01-> Fila Remota App01-> Fila Local App02 -> MDB App02 -> Mensagem no SystemOut.log App02

#### 4.1. Construção da aplicação de teste de mensageria (BVMessaging.EAR)

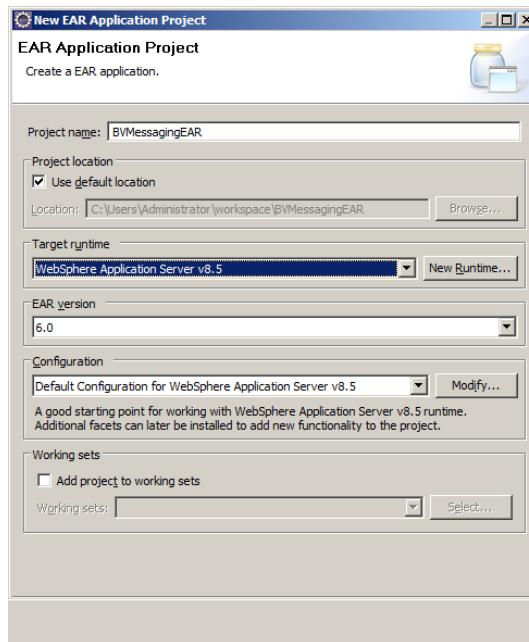
- Abra o Eclipse Luna, cujo atalho está no Desktop do usuário:



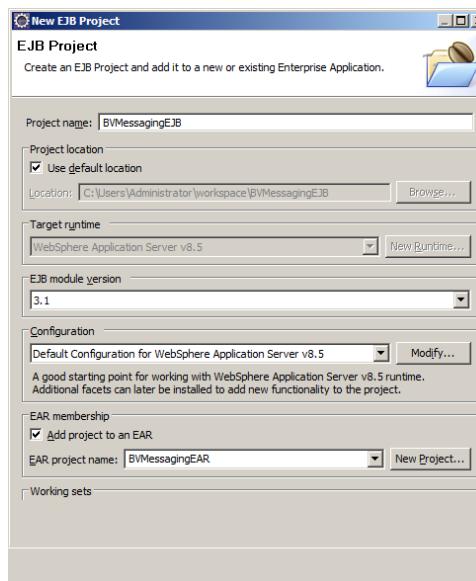
- Crie um novo projeto do tipo Enterprise Application Project:



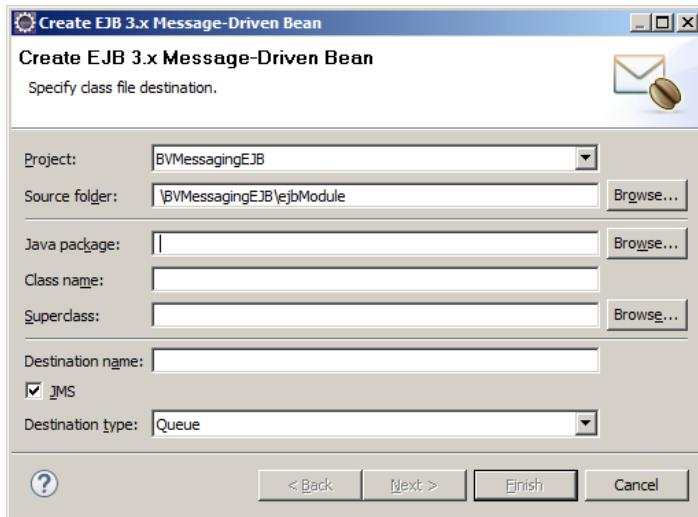
- c) Crie um novo projeto do tipo Enterprise Application Project chamado BVMessagingEAR e com o Target runtime para WebSphere Application Server v8.5:



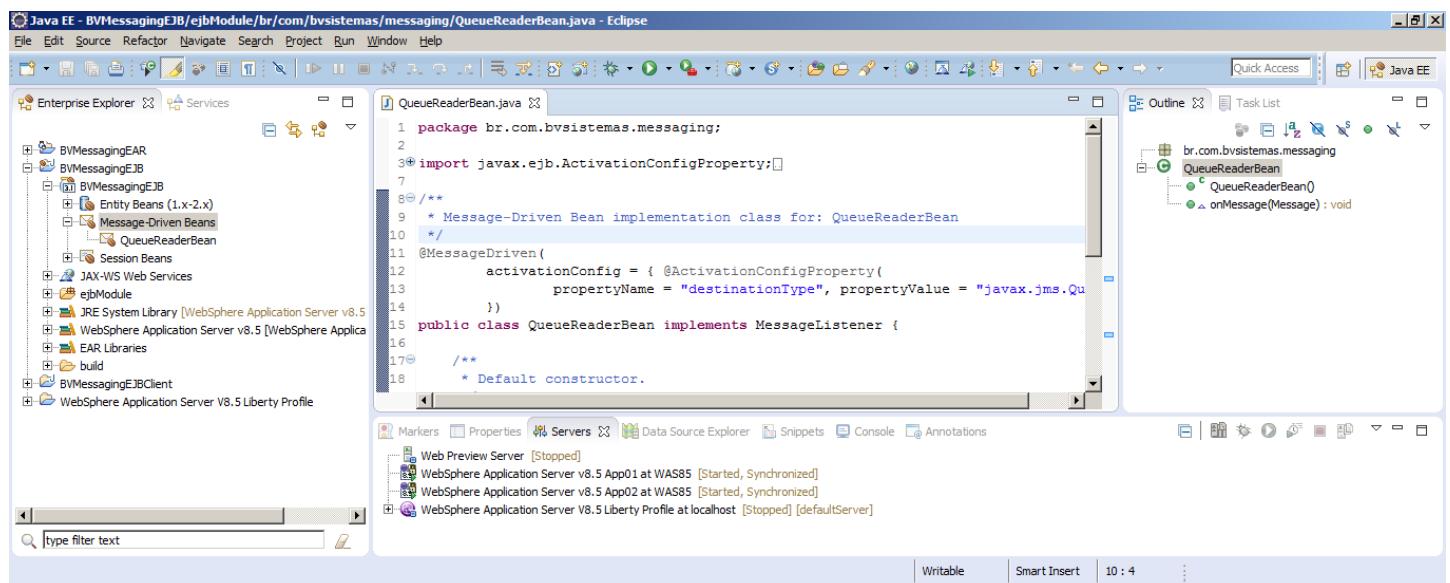
- d) Crie um novo projeto do tipo EJB Project chamado BVMessagingEJB associado ao projeto BVMessagingEAR:



e) No projeto BVMessagingEJB, crie um novo Message-Driven Beans:



f) Digite br.com.bvsistemas.messaging no Java package e QueueReaderBean para o Class name e clique Finish:



g) Substitua a implementação da classe por:

```
package br.com.bvsistemas.messaging;

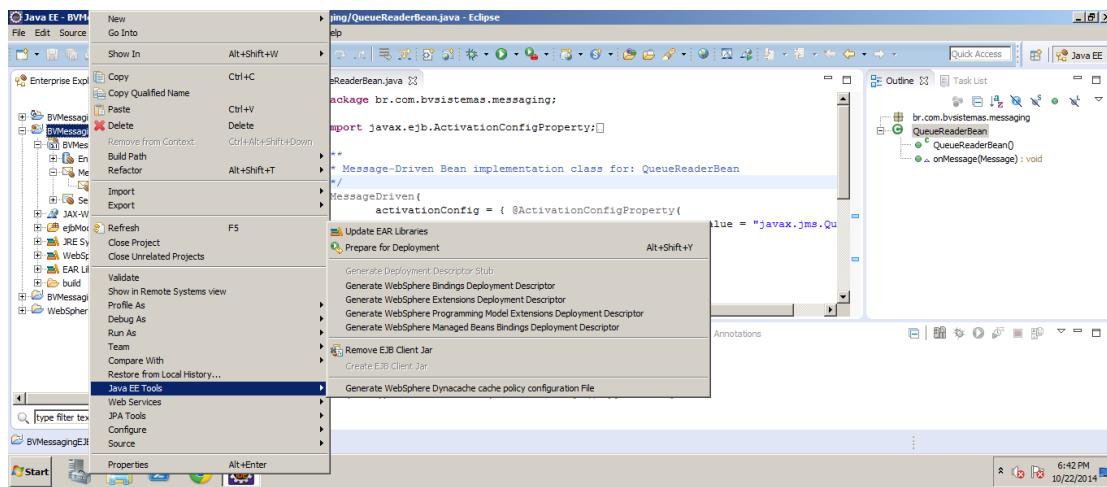
import javax.ejb.MessageDriven;
import javax.jms.JMSEException;
import javax.jms.Message;
import javax.jms.MessageListener;
import javax.jms.TextMessage;

/**
 * Message-Driven Bean implementation class for: QueueReaderBean
 */
@MessageDriven(name="QueueReaderBean")
public class QueueReaderBean implements MessageListener {

    /**
     * Default constructor.
     */
    public QueueReaderBean() {
        // TODO Auto-generated constructor stub
    }

    /**
     * @see MessageListener#onMessage(Message)
     */
    // TODO Auto-generated method stub
    public void onMessage(Message message) {
        try {
            System.out.println("### onMessage(): " + ((TextMessage) message).getText());
        } catch (JMSEException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
}
```

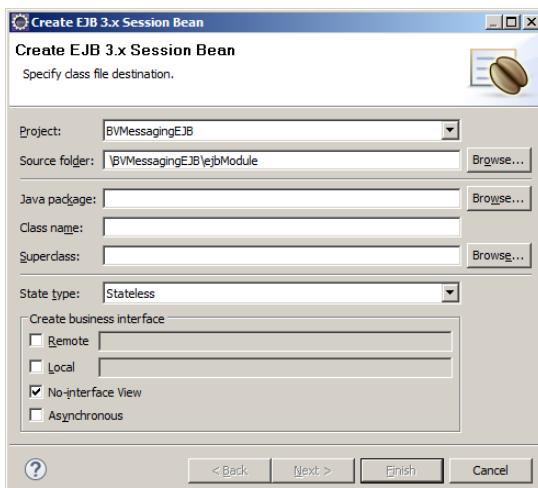
- h) Clique com o botão direto sobre o projeto BVMessagingEJB e selecione a opção Java EE Tools ->Generate WebSphere Bindings Deployment Descriptor:



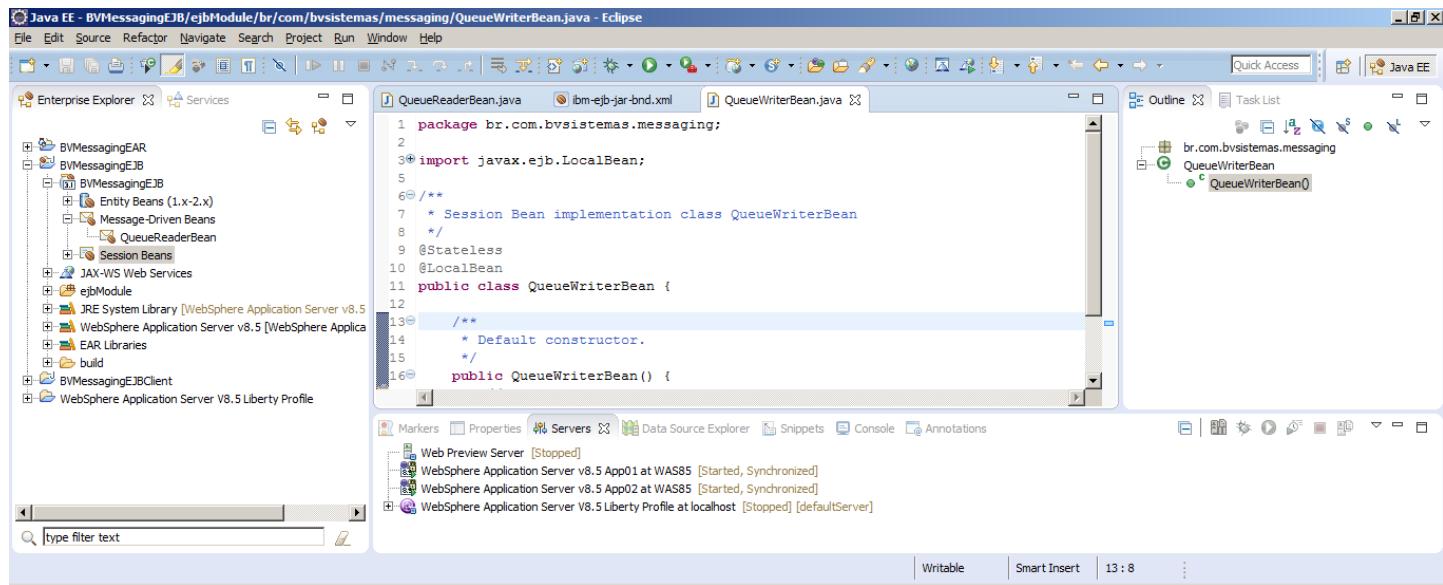
- i) No arquivo ibm-ejb-jar-bnd.xml, substitua o conteúdo do arquivo por:

```
<?xml version="1.0" encoding="UTF-8"?>
<ejb-jar-bnd
    xmlns="http://websphere.ibm.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://websphere.ibm.com/xml/ns/javaee
http://websphere.ibm.com/xml/ns/javaee/ibm-ejb-jar-bnd_1_2.xsd"
    version="1.2">
<message-driven name="QueueReaderBean">
    <jca-adapter activation-spec-binding-name="jms/APPActivationSpec"/>
</message-driven>
</ejb-jar-bnd>
```

- j) Crie um novo Session Bean:



- k) Digite br.com.bvsistemas.messaging no Java package e QueueWriterBean para o Class name e clique Finish:



- l) Substitua a implementação da classe por:

```

package br.com.bvsistemas.messaging;

import javax.annotation.Resource;
import javax.ejb.LocalBean;
import javax.ejb.Stateless;
import javax.jms.ConnectionFactory;
import javax.jms.JMSException;
import javax.jms.Queue;
import javax.jms.QueueConnection;
import javax.jms.QueueConnectionFactory;
import javax.jms.QueueSender;
import javax.jms.QueueSession;
import javax.jms.Session;
import javax.jms.TextMessage;
import javax.jws.WebMethod;
import javax.jws.WebParam;
import javax.jws.WebService;

/**
 * Session Bean implementation class QueueWriterBean
 */
@Stateless
@LocalBean
@WebService

```

```
public class QueueWriterBean {  
  
    @Resource(lookup = "jms/QCF")  
    private ConnectionFactory connFactory;  
  
    @Resource(lookup = "jms/INTEGRATION.REQUEST")  
    private Queue queue;  
  
    /**  
     * Default constructor.  
     */  
    public QueueWriterBean() {  
        // TODO Auto-generated constructor stub  
    }  
  
    @WebMethod  
    public String helloWorld(@WebParam(name="phrase") String phrase){  
        QueueConnection connection = null;  
        QueueSession session = null;  
  
        try {  
            QueueConnectionFactory factory = (QueueConnectionFactory) connFactory;  
            connection = factory.createQueueConnection();  
            session = connection.createQueueSession(false,  
                Session.AUTO_ACKNOWLEDGE);  
            QueueSender sender = session.createSender(queue);  
            TextMessage textMessage = session.createTextMessage();  
  
            textMessage.setText(phrase);  
            sender.send(textMessage);  
            connection.close();  
  
            System.out.println(textMessage.getJMSMessageID());  
        } catch (JMSEException jmse) {  
            jmse.printStackTrace();  
        } finally {  
            if (connection != null) {  
                try {  
                    connection.close();  
                } catch (JMSEException jmse) {  
                    jmse.printStackTrace();  
                }  
            }  
        }  
    }  
}
```

```

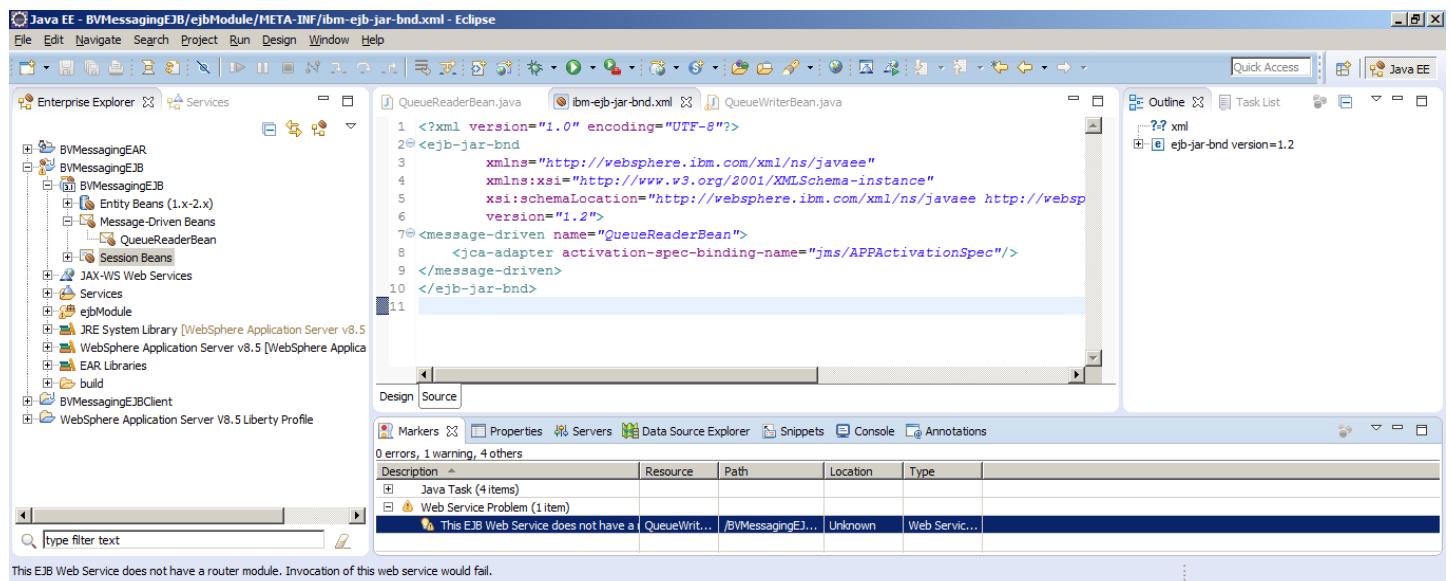
    }

    return phrase;
}

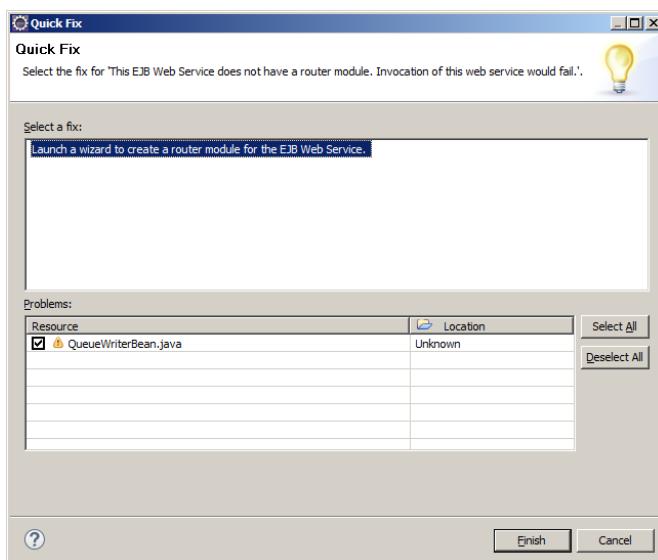
}

```

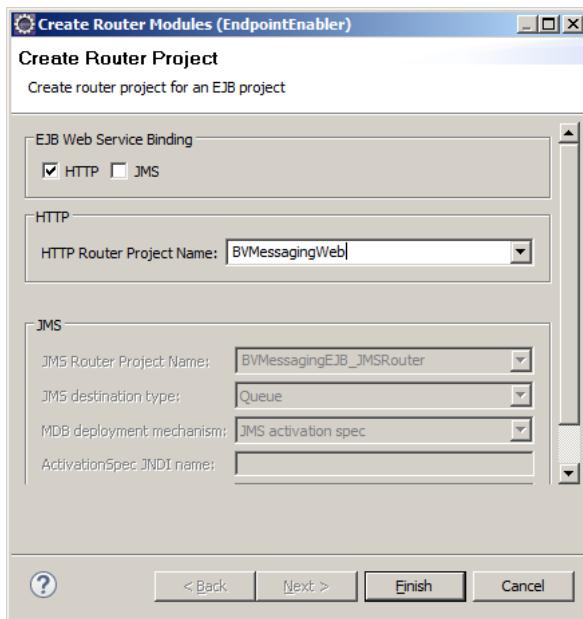
m) Clique na view Markers e expanda o problema Web Service Problem:



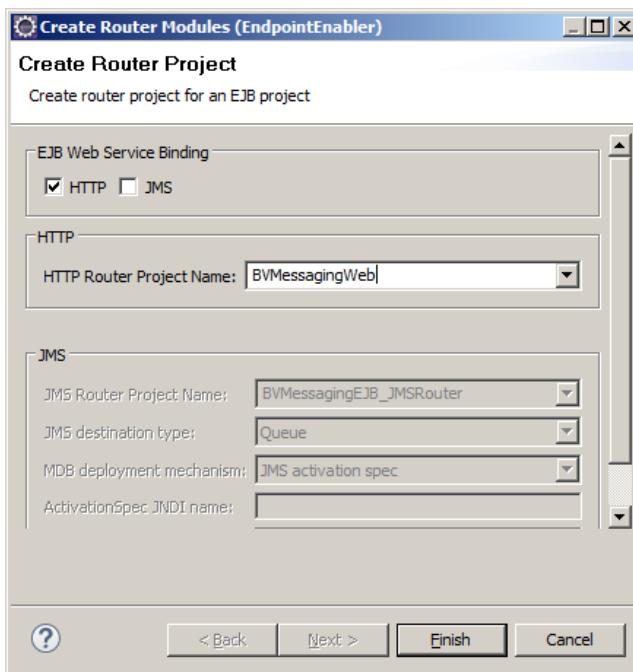
n) Clique com o botão direito em cima do problema e selecione Quick Fix:



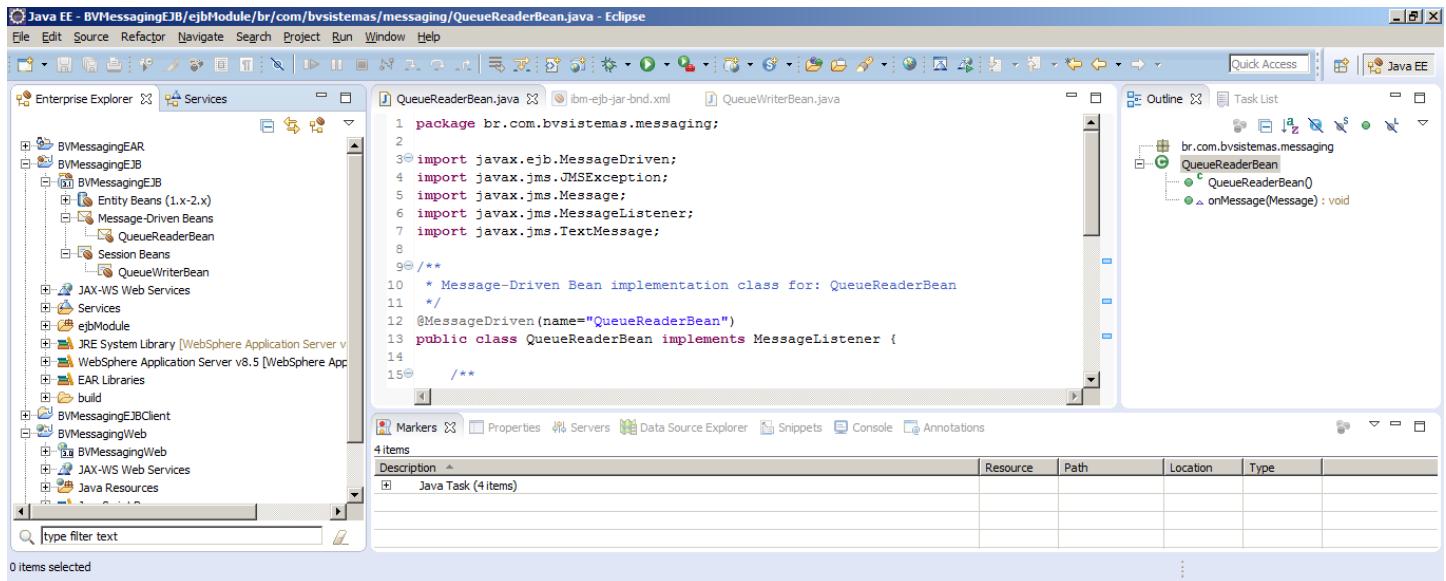
o) Clique Finish para lançar o Wizard de criação do WebService:



p) Digite BVMessagingWeb para o HTTP Router Project Name e clique Finish:



q) Os artefatos foram criados com sucesso:



The screenshot shows the Eclipse IDE interface for Java EE development. The title bar reads "Java EE - BVMessagingEJB/ejbModule/br/com/bvsistemas/messaging/QueueReaderBean.java - Eclipse". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar has various icons for file operations like Open, Save, Cut, Copy, Paste, Find, and Refresh. The left-hand Enterprise Explorer view shows the project structure under "BVMessagingEAR": BVMessagingEJB (with Entity Beans (1.x-2.x), Message-Driven Beans (QueueReaderBean), Session Beans (QueueWriterBean)), JAX-WS Web Services, Services, ejbModule, JRE System Library [WebSphere Application Server v8.5], WebSphere Application Server v8.5 [WebSphere App], EAR Libraries, build, and BVMessagingEJBClient (with BVMessagingWeb, JAX-WS Web Services, Java Resources). The center workspace displays the code for "QueueReaderBean.java":

```
1 package br.com.bvsistemas.messaging;
2
3 import javax.ejb.MessageDriven;
4 import javax.jms.JMSException;
5 import javax.jms.Message;
6 import javax.jms.MessageListener;
7 import javax.jms.TextMessage;
8
9 /**
10  * Message-Driven Bean implementation class for: QueueReaderBean
11  */
12 @MessageDriven(name="QueueReaderBean")
13 public class QueueReaderBean implements MessageListener {
14
15 }
```

The right-hand Outline view shows the class structure: "br.com.bvsistemas.messaging.QueueReaderBean" with a method "onMessage(Message) : void". Below the workspace are several toolbars: Markers, Properties, Servers, Data Source Explorer, Snippets, Console, and Annotations. The bottom status bar indicates "4 items selected".





### 5. Instalação da aplicação de teste da mensageria

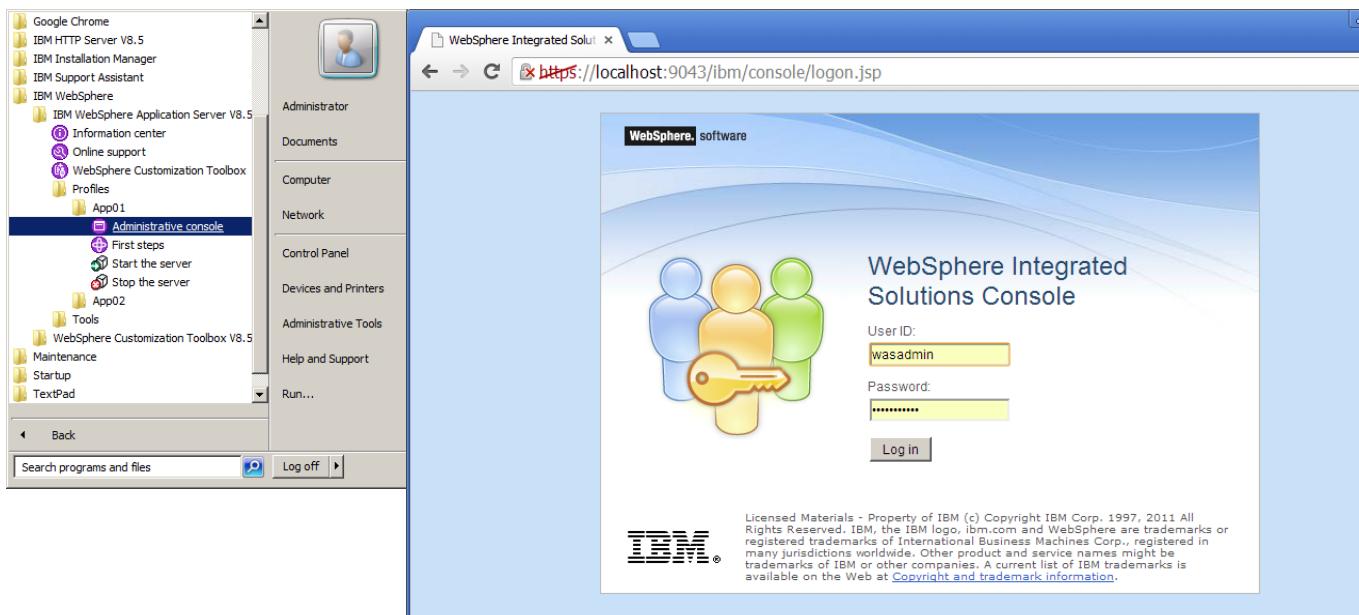
A aplicação de teste de mensageria é composta por um Message Driven Bean que lê da fila local do Service Integration Bus e de um Session Bean, que escreve na fila remota do SIB. O Session Bean está exposto como um Web Service, logo utilizaremos o SOAPUI para testar a chamada a essa requisição.

O fluxo da informação a ser validado é o seguinte:

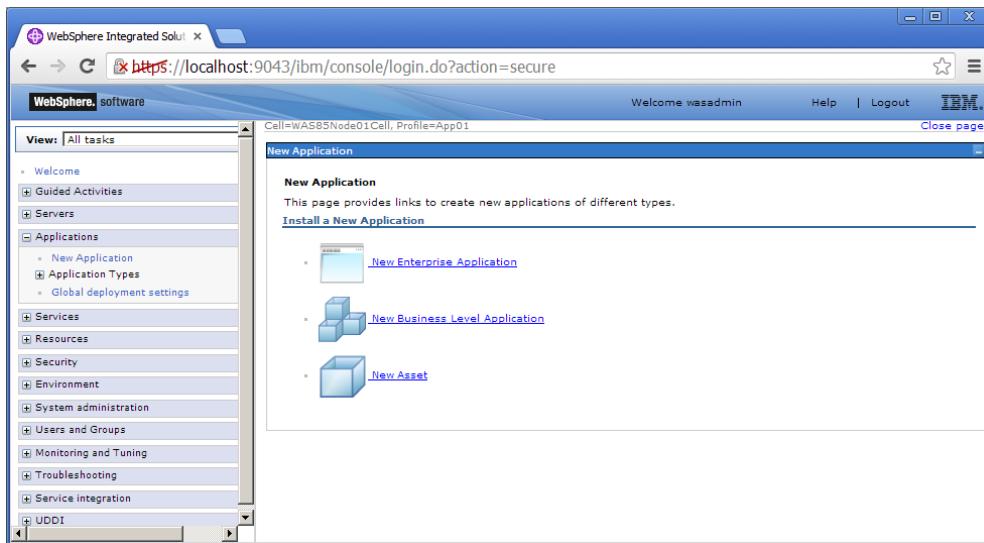
Mensagem no SOAPUI -> WebService App01 -> Session Bean App01-> Fila Remota App01-> Fila Local App02 -> MDB App02 -> Mensagem no SystemOut.log App02

#### 5.1. Instalação da aplicação de teste (BVMessaging.EAR)

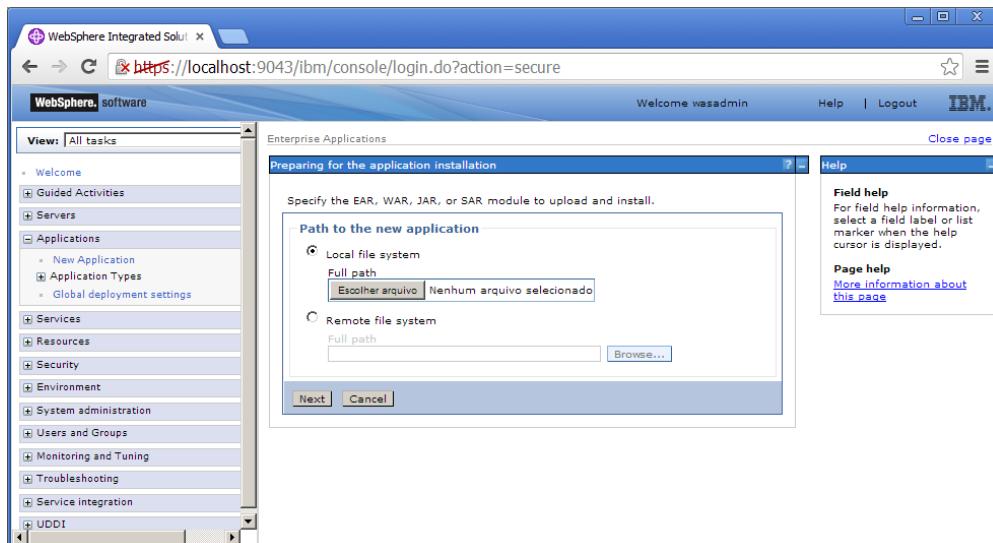
- s) Instale a aplicação no App01. Abra a console administrativa do App01 (<http://localhost:9060/admin>) ou pelo atalho:



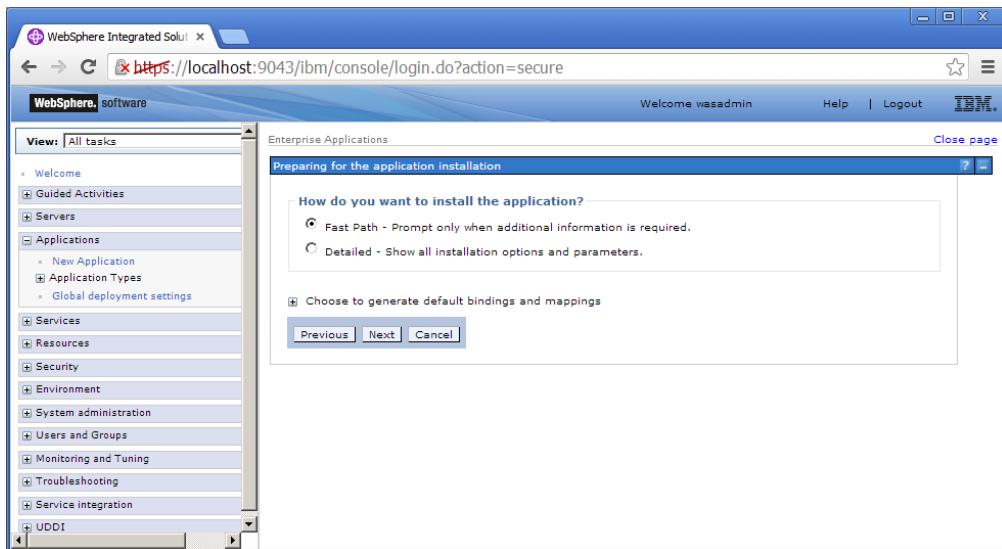
t) Na console administrativa, expanda Applications e clique em New Application:



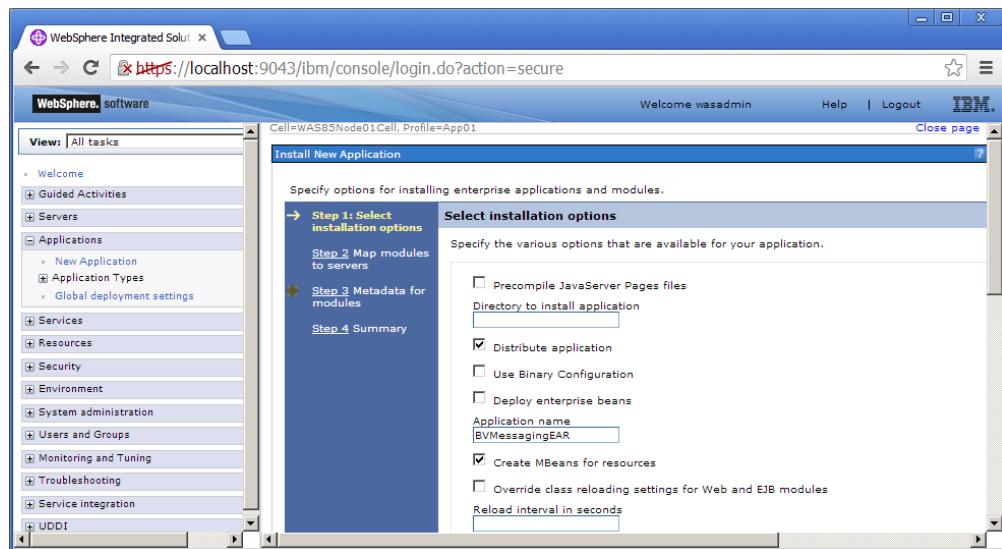
u) Selecione New Enterprise Application:



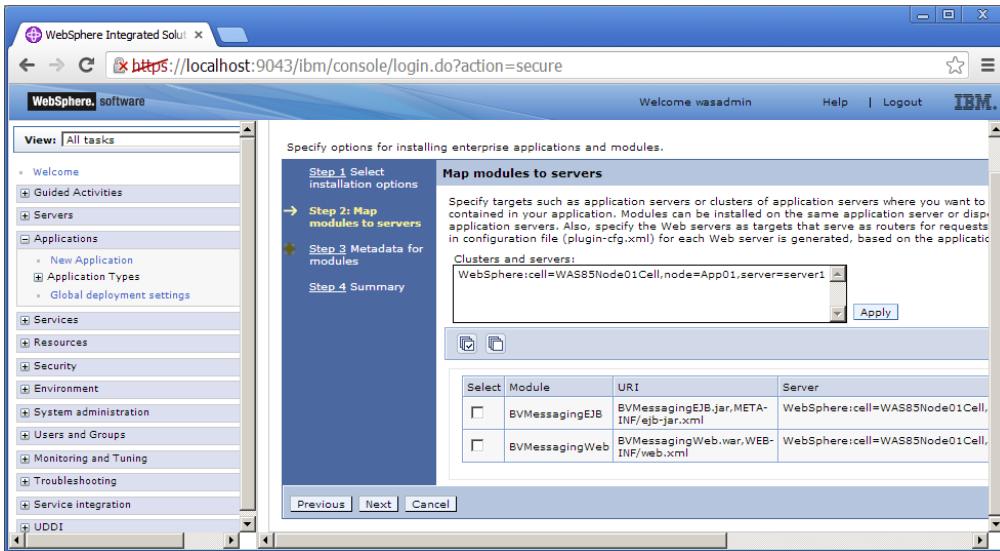
v) Escolha o arquivo BVMessaging.ear localizado no Desktop do usuário e clique Next:



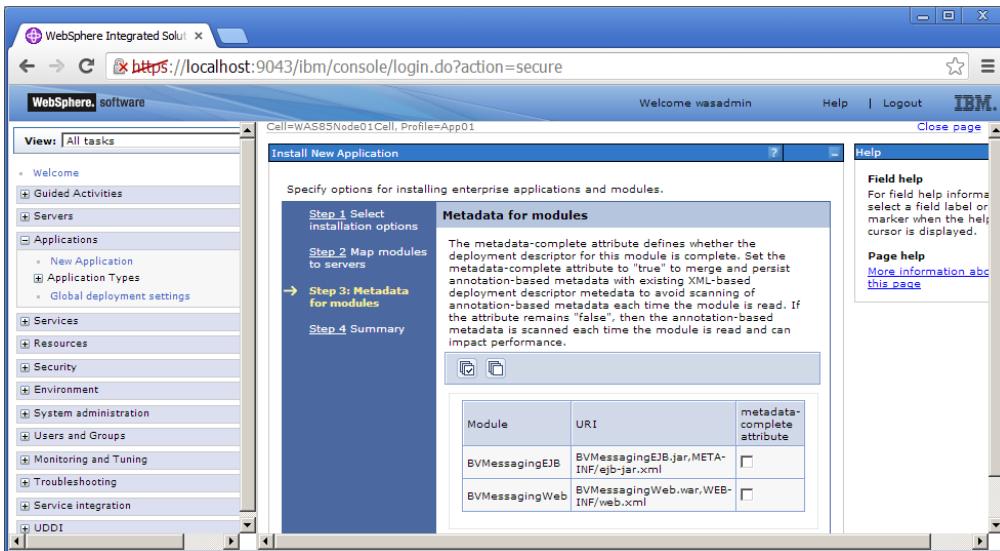
w) Aceite as configurações Default e clique Next:



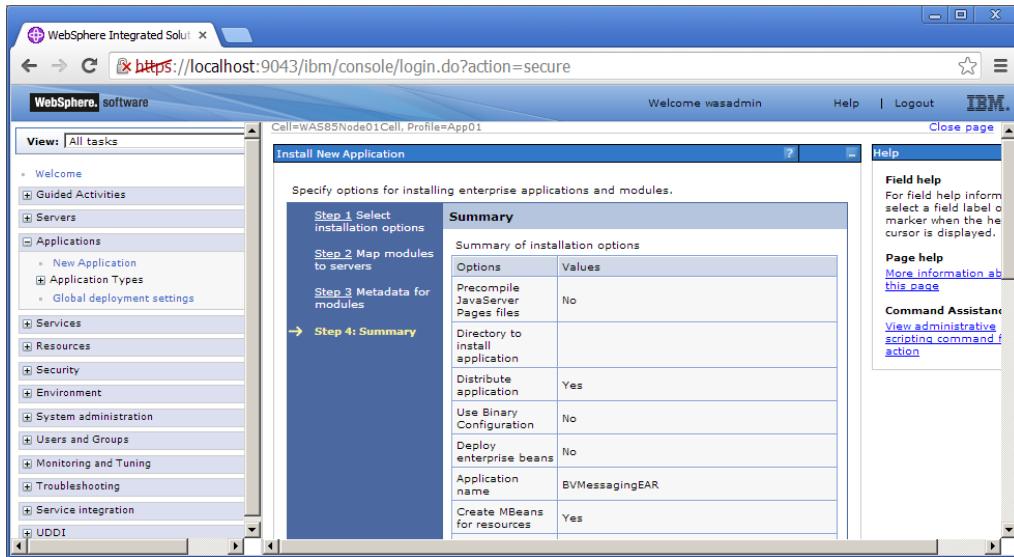
x) Aceite as configurações Default e clique Next:



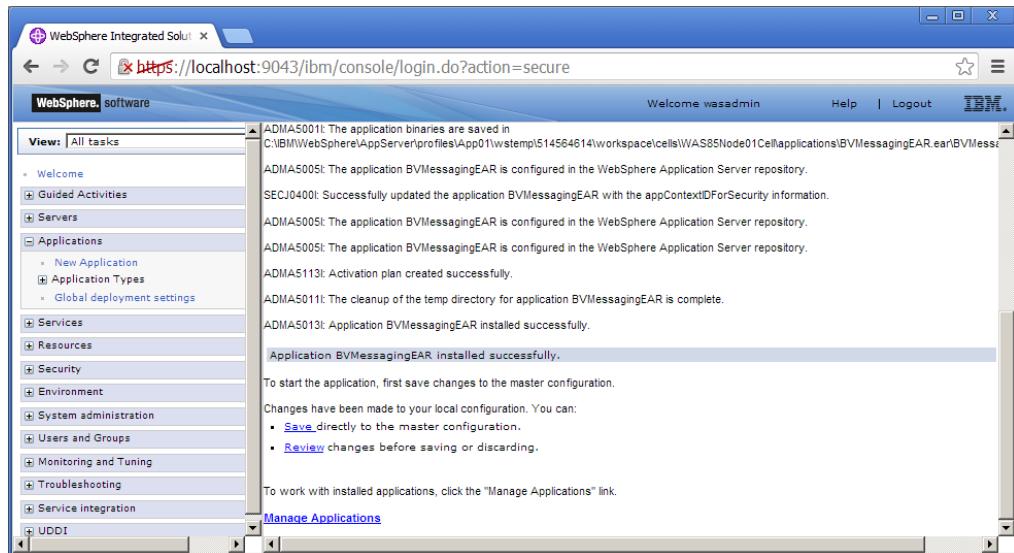
y) Aceite as configurações Default e clique Next:



z) Reveja as configurações e clique Finish:

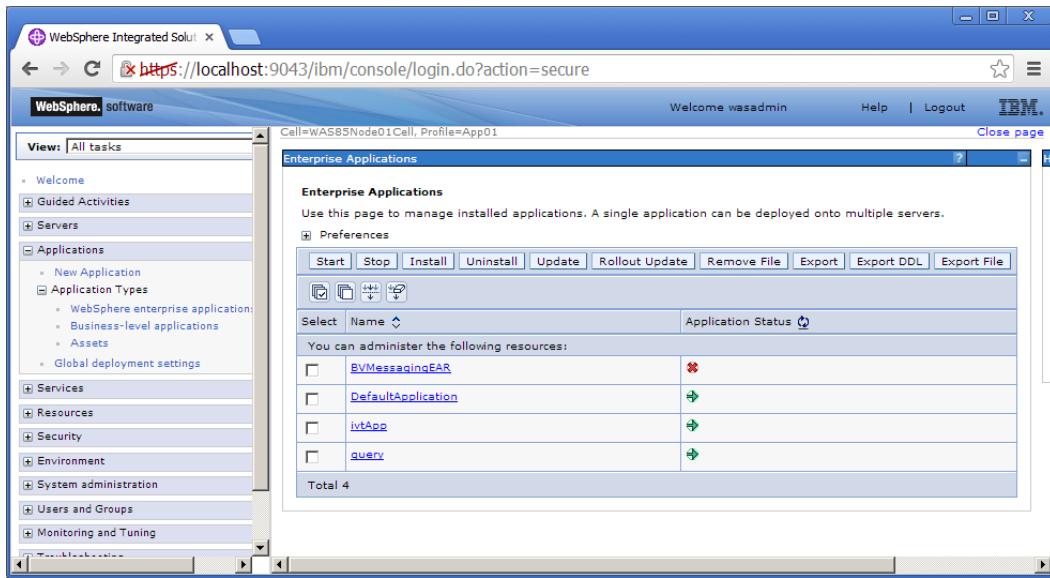


aa) Salve as configurações na Master Configuration:

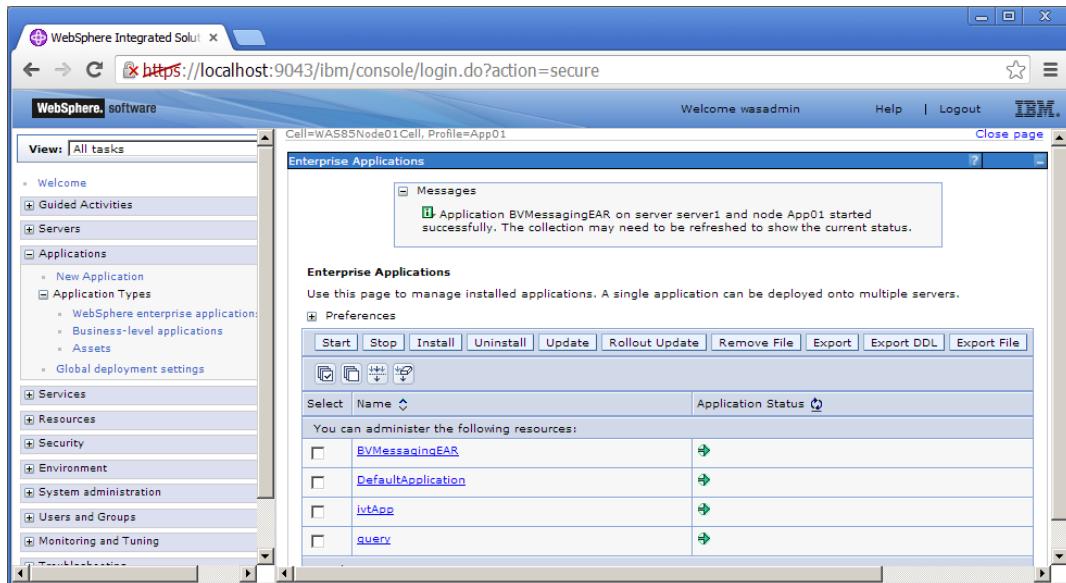


bb) Repita o procedimento de instalação da aplicação no App02.

cc) Na console do App01, expanda Applications->Application Types e clique em WebSphere enterprise applications:



dd) Selecione a aplicação BVMessagingEAR e clique no botão Start para iniciar a aplicação:

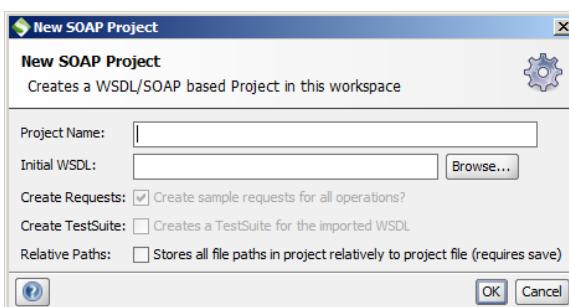


### 5.2. Testes da aplicação

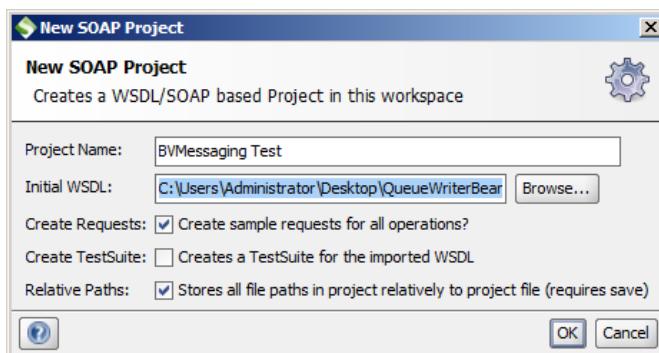
- a) Abra o SOAPUI para fazer a chamada do WebService:



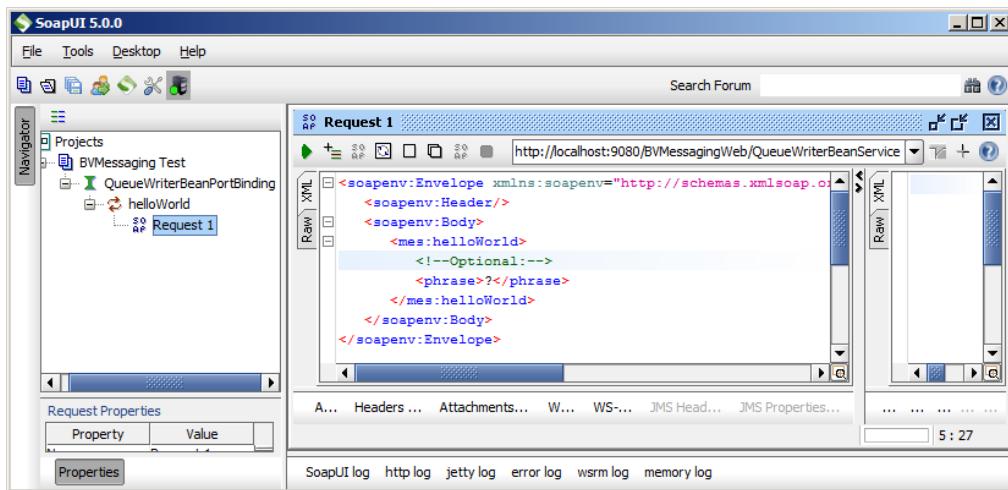
- b) Clique com o botão direito em Projects e selecione New SOAP Project:



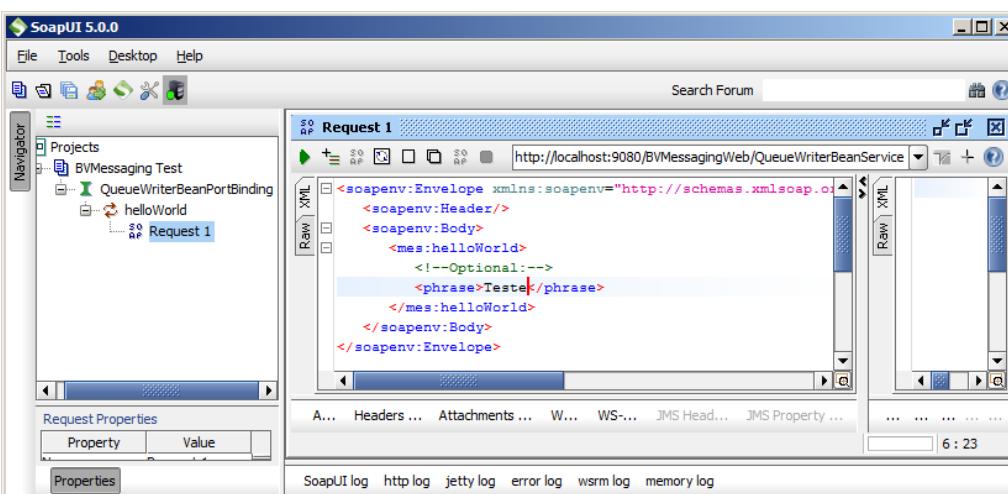
- c) Digite  
BVMessaging Test em Project Name e  
C:\Users\Administrator\Desktop\QueueWriterBeanService.wsdl em Initial WSDL:



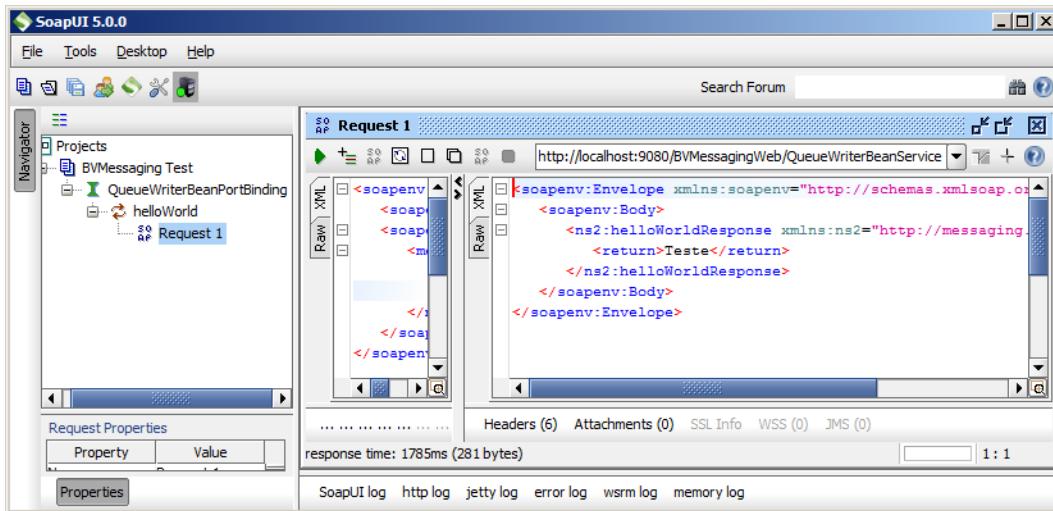
d) Clique OK para gerar o projeto:



e) Expanda BVMessaging Test->QueueWriterBeanPortBinding->helloWorld->Request1 e preencha a string a ser enviada no WebService:



- f) Clique no botão Play para enviar a requisição:

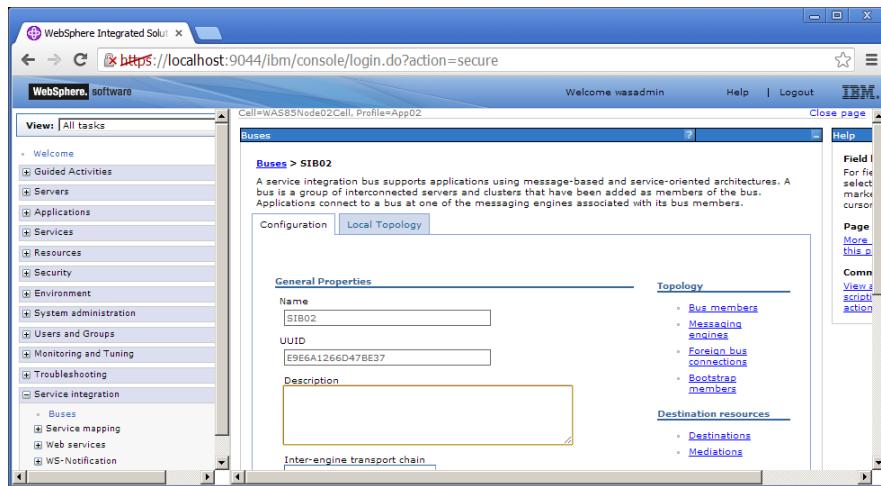


- g) Neste momento, o Session Bean colocou uma mensagem na fila SIB02.REQ.

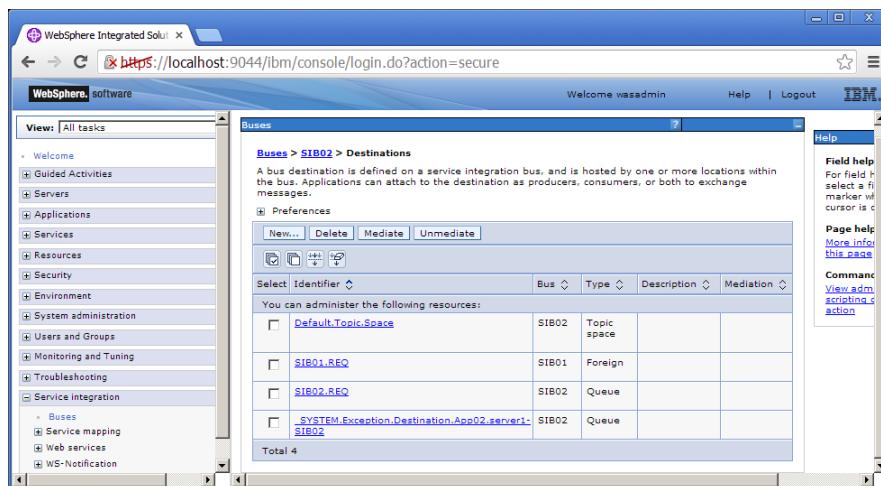
- h) Entre na console administrativa do App02:



- i) Expanda Service integration e clique em Buses. Clique em SIB02:

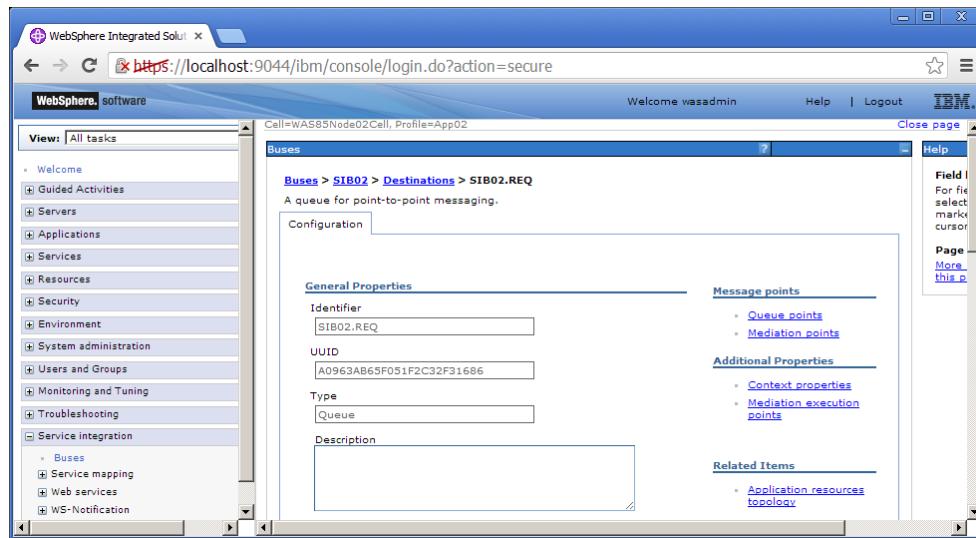


- j) Clique em Destination resources->Destinations:

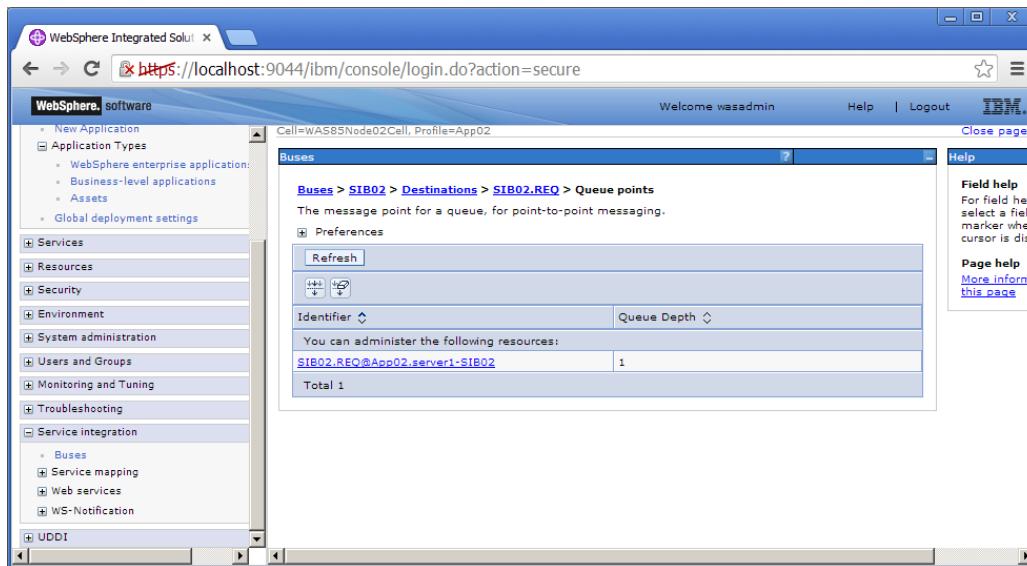


Identifier	Bus	Type	Description	Mediation
Default_Topic_Space	SIB02	Topic space		
SIB01_REQ	SIB01	Foreign		
SIB02_REQ	SIB02	Queue		
SYSTEM.Exception.Destination.App02.server1-SIB02	SIB02	Queue		

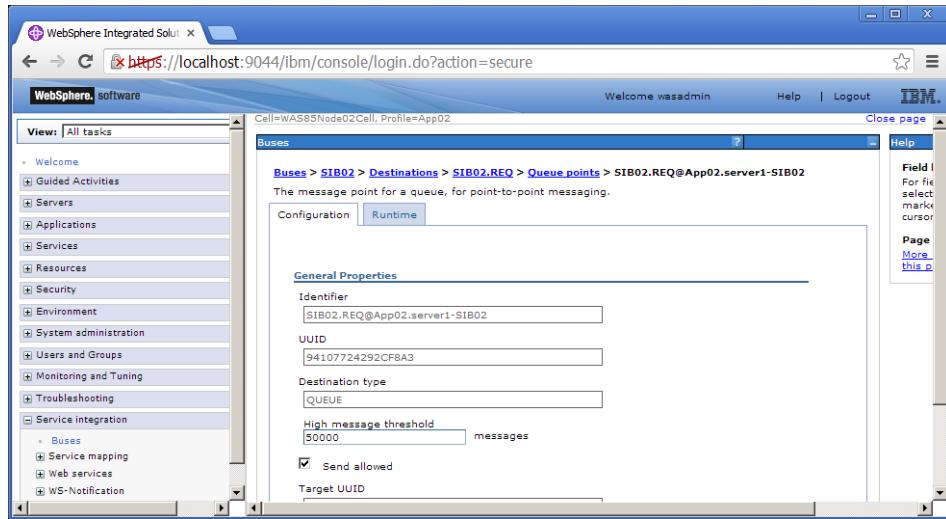
k) Clique em SIB02.REQ:



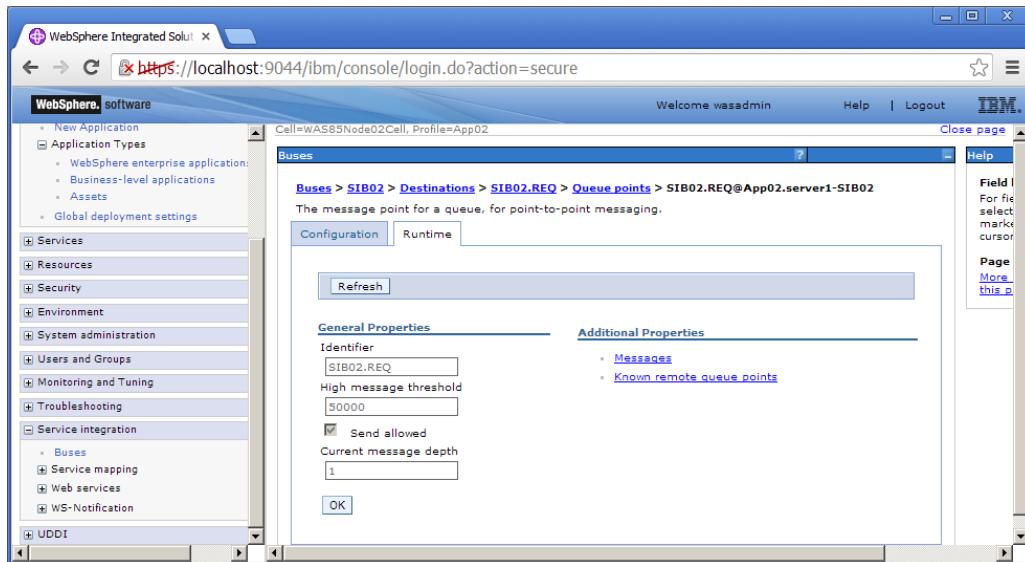
l) Clique em Message Points-> Queue points:



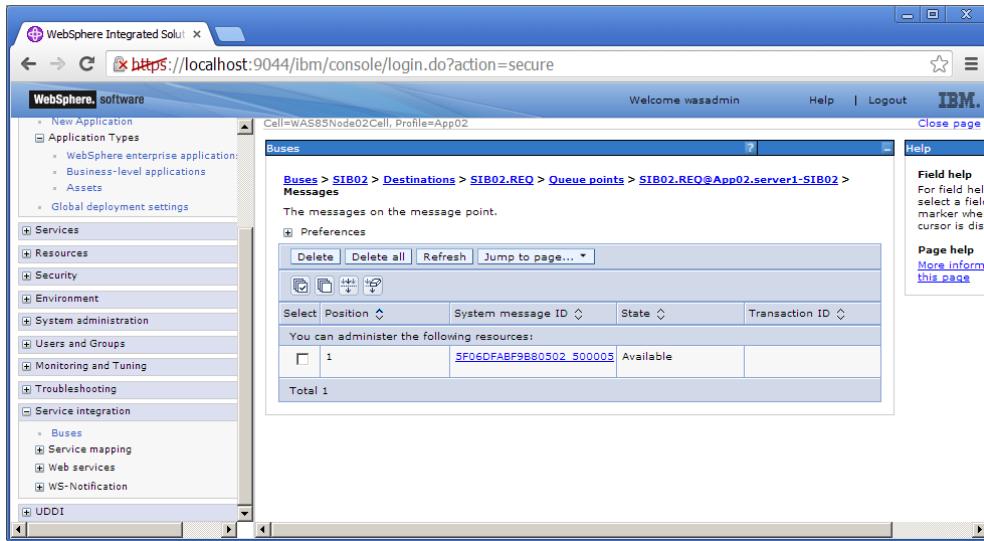
m) Clique em SIB02.REQ@App02.server1-SIB02:



n) Selecione a aba Runtime:

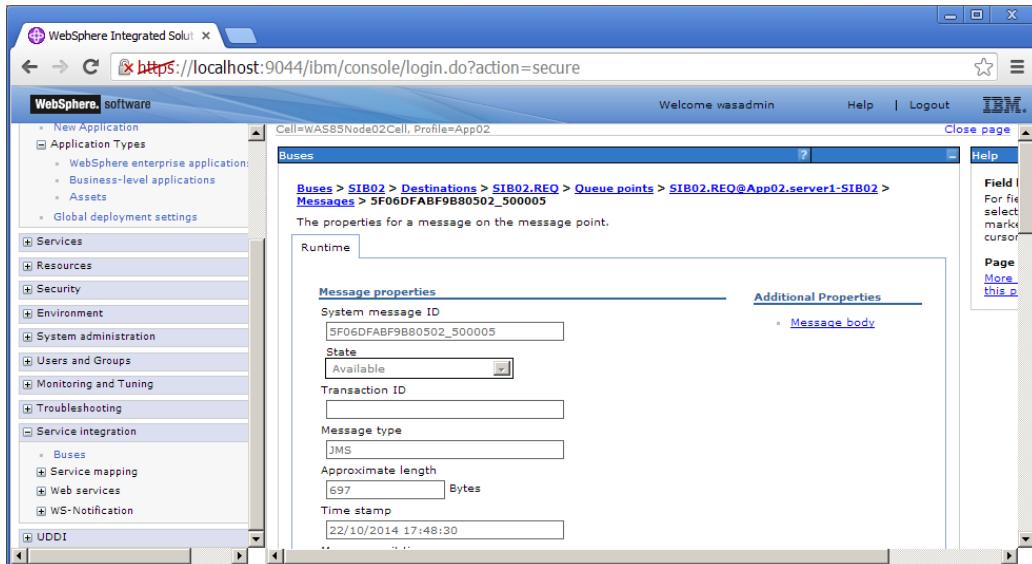


o) Clique em Messages:

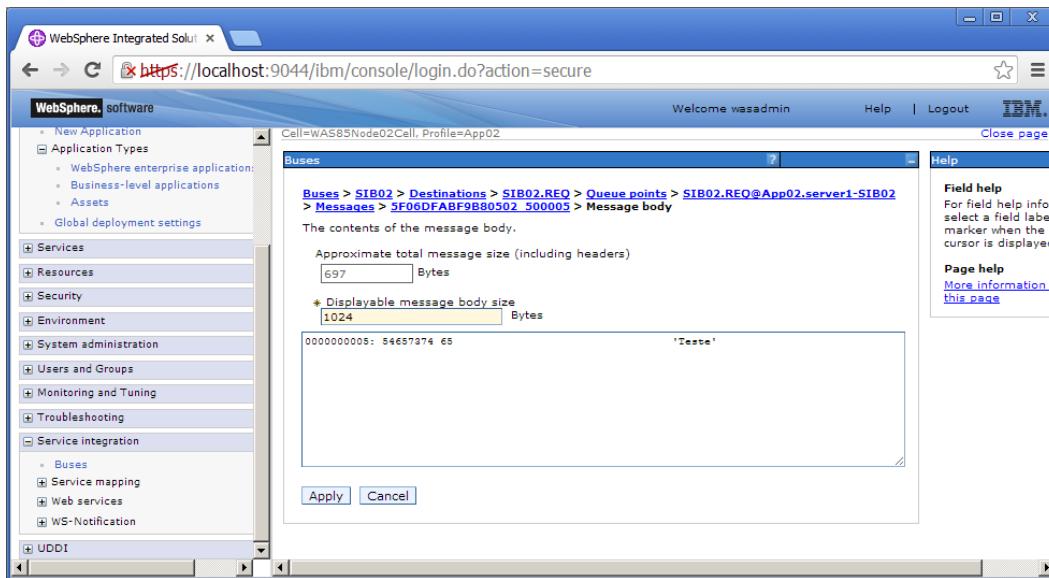


Select	Position	System message ID	State	Transaction ID
<input type="checkbox"/>	1	SF06DFABF9B80502_500005	Available	

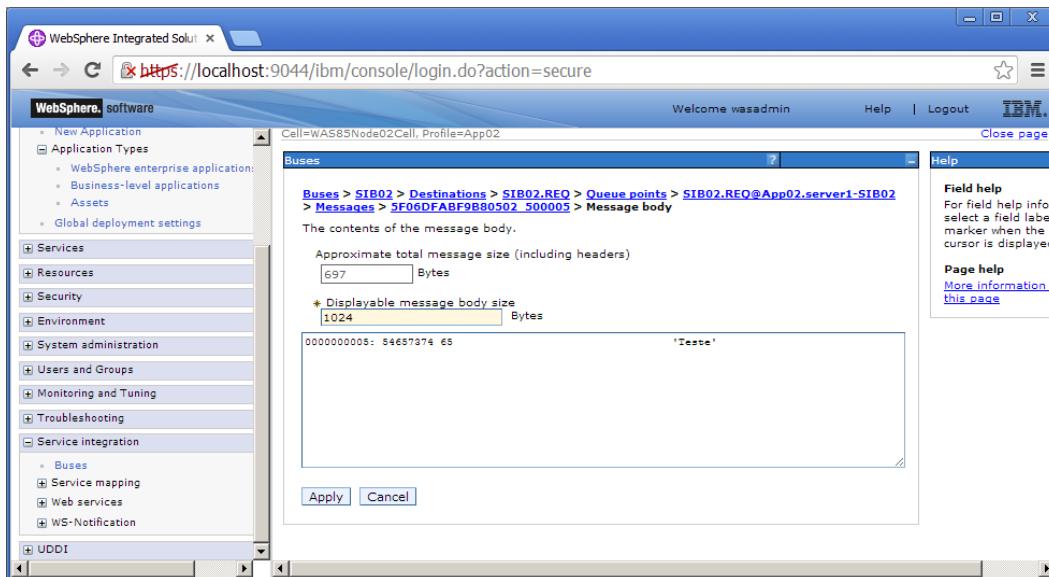
p) Clique no link da mensagem:



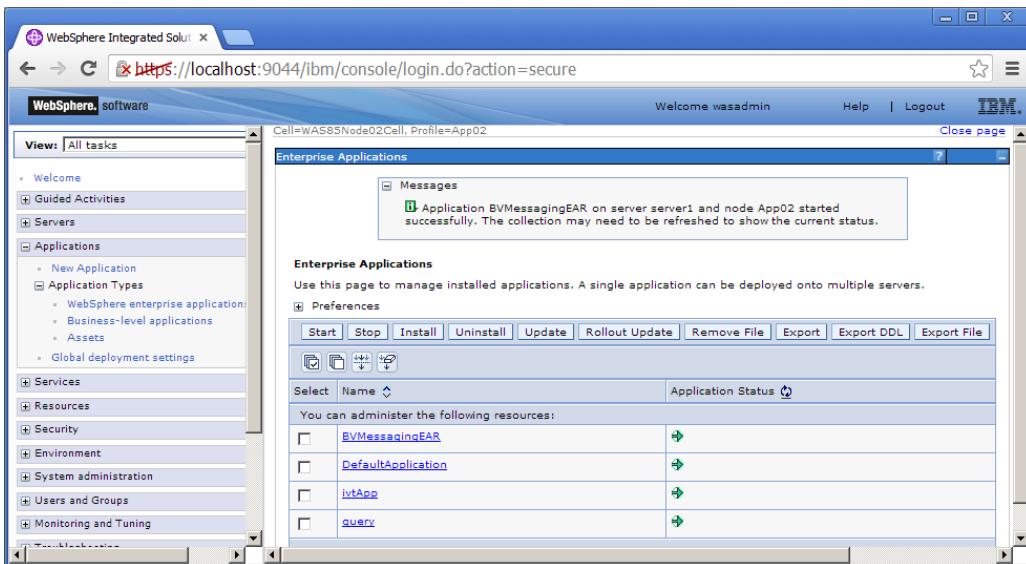
q) Clique em Additional Properties->Message Body para visualizar o corpo da mensagem:



r) Clique em Additional Properties->Message Body para visualizar o corpo da mensagem:



s) Inicie a aplicação BVMessagingEAR no App02:



t) Veja a saída do SystemOut.log do App02 localizado em C:\IBM\WebSphere\AppServer\profiles\App02\logs\server1\SystemOut.log:

```
[10/22/14 18:15:28:097 BRST] 0000007b ApplicationMg A WSVR0221I: Application started:  
BVMessagingEAR  
[10/22/14 18:15:28:098 BRST] 0000007b CompositionUn A WSVR0191I: Composition unit  
WebSphere:cuname=BVMessagingEAR in BLA WebSphere:blaname=BVMessagingEAR started.  
[10/22/14 18:15:28:148 BRST] 000000c0 SystemOut O ### onMessage(): Teste
```