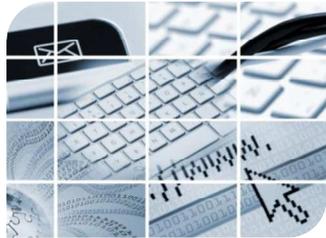




Instalação do Server – Intellector Inteligência de Crédito

## Intellector – Instalação do Server

### Programa do Treinamento



#### I) Objetivo do Treinamento

#### II) Introdução

#### III) JDK 6.0

#### IV) JBoss

#### V) Intellector-EAR

#### VI) Configurando a "fila" do Intellector

#### VII) Persistence.properties

2

Olá, antes de iniciarmos, é importante reforçar alguns pontos sobre o treinamento, seus objetivos e conteúdo programático.

Nosso programa compreende as seguintes etapas:

- Objetivo do Treinamento
- Introdução
- JDK 6.0
- JBoss
- Intellector-EAR
- Configurando a "fila" do Intellector
- Persistence.properties

## Intellector – Instalação do Server

### Objetivo do Treinamento



Orientar o processo de instalação do servidor Intellector.

3

### Objetivo do Treinamento

O objetivo principal deste treinamento é orientar o processo de instalação do servidor Intellector.

Esperamos levar até você informações relevantes que os auxiliem em seu trabalho, buscando esclarecer as principais questões referentes à **Instalação do Server**.  
Uma ótima aula para você!

## Intellector – Instalação do Server

### Introdução

Este treinamento tem o objetivo de orientar quanto à instalação do servidor Intellector para o JBoss Application Server.

Para esta tarefa é necessário prévio conhecimento de JBoss e JVM.

4

### Introdução

Este treinamento tem o objetivo de orientar quanto à instalação do servidor Intellector para o JBoss Application Server.

O público alvo para este treinamento são administradores de sistemas e application server, que tenham bom conhecimento de JBoss e JVM. Portanto, este treinamento assume que os administradores tem familiaridade com os conceitos de administração e deployment do JBoss, Linux ou Windows e Sun JDK.

**Intellector – Instalação do Server**

**JDK 6.0**

- Instale a Sun JDK 6.0 apropriada para sua plataforma. As JDKs estão disponíveis no seguinte site: <http://java.sun.com/javase/downloads/index.jsp>
- A versão atual do Intellector **SOMENTE** funciona com a **Java SE Development Kit (JDK) 6 Update 10** ou superior.
- Criar no Windows uma pasta chamada “opt” na raiz do disco local ou usar o '/opt' no Unix. Instale o JDK nesta pasta.
- Atenção as permissões de escrita para o diretório JBoss em sistemas Unix pois o Intellector precisará escrever nesse diretório.

5

## JDK 6.0

Instale a Sun JDK 6.0 apropriada para sua plataforma. As JDKs estão disponíveis no site mostrado no slide. <http://java.sun.com/javase/downloads/index.jsp>

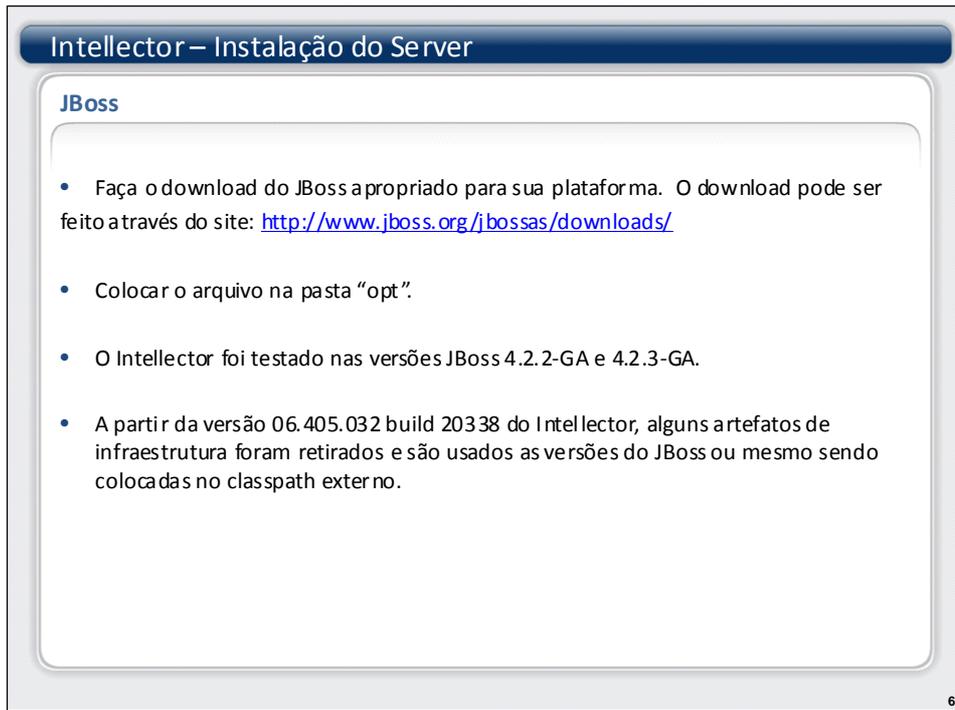
Recomendamos enfaticamente, usar a JDK superior ao Update 10. O Intellector **NÃO FUNCIONA** com JDK inferior ao Update 7 devido à API do JAXB.

Não é suficiente ter somente o Java Runtime Environment (JRE); você realmente precisa do full-blown JDK.

Forneceremos um bootstrap para o Intellector, onde a variável `JAVA_HOME` é configurada, mas caso você queira colocá-la em outro lugar, não se esqueça de configurá-la no seu ambiente.

Embora a localização da instalação da JDK seja livre, sugerimos criar no Windows uma pasta chamada “opt” na raiz do disco local, ou usar o '/opt' no Unix. Ajuda para propósitos de suporte.

Instalando em sistemas Unix, tenha bastante cuidado com as permissões de escrita para owner/group para o diretório do JBoss, pois o Intellector vai precisar escrever nesse diretório.



## JBoss

Faça o download do JBoss apropriado para sua plataforma.

<http://www.jboss.org/jbossas/downloads/>

Colocar o arquivo na pasta "opt".

O download pode ser feito através do site mostrado no slide.

O Intellector foi testado nas versões JBoss 4.2.2-GA e 4.2.3-GA.

A partir da versão 06.405.032 build 20338 do Intellector, alguns artefatos de infraestrutura foram retirados do seu WEB-INF/lib e usado as versões do JBoss ou mesmo sendo colocadas no classpath externo (\$JBASS\_HOME/server/[instancia]/lib)

Temos duas opções para colocar no classpath essas bibliotecas de infraestrutura:

a) simplesmente copiando para a pasta \$JBASS\_HOME/server/[instancia]/lib

b) criando uma pasta na instância, copiando os artefatos para ela (ex.: "intellector-lib") e alterando o arquivo de configuração \$JBASS\_HOME/server/[instancia]/conf/jboss-service.xml, adicionando as linhas como abaixo:

logo abaixo de:

```
<classpath codebase="${jboss.server.lib.url:lib}" archives="*" />
```

acrescente....

```
<classpath codebase="intellector-lib" archives="*" />
```

Artefatos neste tarball e seus respectivos links de download, se houverem:

a) Apache CXF - implementa JAX-WS (web services)

\* cxf-2.7.0.jar

link: <http://cxf.apache.org/download.html>

b) Hibernate - persistencia, implementa JPA

\* hibernate3.jar

link: <http://sourceforge.net/projects/hibernate/files/hibernate3/3.6.10.Final/>

c) Logging usado pelo hibernate

\* slf4j-api-1.5.8.jar

\* slf4j-jcl-1.5.8.jar

d) Spring Core Framwork - injeção de dependencia (Transport/access/proxy)

\* spring-beans-2.5.6.jar

\* spring-context-2.5.6.jar

\* spring-core-2.5.6.jar

\* spring-web-2.5.6.jar

<http://sourceforge.net/projects/springframework/files/springframework-2/2.5.6/>



### Teste de verificação da instalação

Após instalar a Java JDK e o JBoss nos locais apropriados (na pasta opt), faça um pequeno teste de verificação da instalação. Para verificar se as instalações ocorreram perfeitamente, primeiro execute o JBoss, e depois, acesse a url mostrada no slide (<http://localhost:8080/web-console>).

Para executar o JBoss pelo Windows, acesse a pasta onde estão os arquivos, subpasta /Bin e execute o arquivo run.bat.

No caso de ambiente Unix/Linux, abra um terminal e execute o arquivo run.sh.

Se tudo ocorreu bem até esse momento, você estará apto a visualizar a página de administração do application server.

## Intellector – Instalação do Server

### JBoss – Bootstrap para JBoss/Intellector

- **No Linux/Unix:** Insira o Script de bootstrap `run-intellector.sh` em `<JBOSS_HOME>/bin`

```
#!/bin/bash

# alguns snippets para o bootstrap do Intellector
export JAVA_OPTS="-Xms512m -Xmx536m -XX:PermSize=128m -XX:MaxPermSize=128m -Xss128k -Dintellector.datadir=/home/jboss/intellector"

# exportar a variavel JBOSS_HOME, devido a compilacao de politicas
export JBOSS_HOME=/opt/jboss-4.2.3.GA

# a library log4j deve estar incluído no classpath, imprescindível na
# compilacao de politicas; simplesmente não compila as políticas
export JBOSS_CLASSPATH=$JBOSS_HOME/server/default/lib/log4j.jar

# inicia o JBoss numa instancia e IP especificos
./run.sh -c default --host 0.0.0.0
```

8

### Bootstrap para JBoss/Intellector

Os scripts de bootstrap para o Intellector são simples, precisando apenas de pequenas modificações no que concerne a memória e variáveis de ambientes.

O script de bootstrap para Linux/Unix é apresentado no slide.

Para uso no Linux/Unix sugerimos usar o nome 'run-intellector.sh'.

O Script deve ficar em `<JBOSS_HOME>/bin` com permissão explícita de execução.

## Intellector – Instalação do Server

### JBoss – Bootstrap para JBoss/Intellector

- **No Windows** : Insira o Script de bootstrap **run-intellecator.bat** em <JBOSS\_HOME>/bin

```
rem alguns snipets para o bootstrap do Intellector
set JAVA_OPTS=-Xms512m -Xmx536m -XX:PermSize=128m -XX:MaxPermSize=128m -
Xss128k -Dintellecator.datadir=c:/java/intellecator-files

rem exportar a variavel JBOSS_HOME, devido a compilacao de politicas
set JBOSS_HOME=c:\java\jboss-4.2.3.GA

rem a library log4j deve estar incluído no classpath, imprescindível na
rem compilacao de politicas; simplesmente não compila as políticas
set JBOSS_CLASSPATH=%JBOSS_HOME%\server\default\lib\log4j.jar

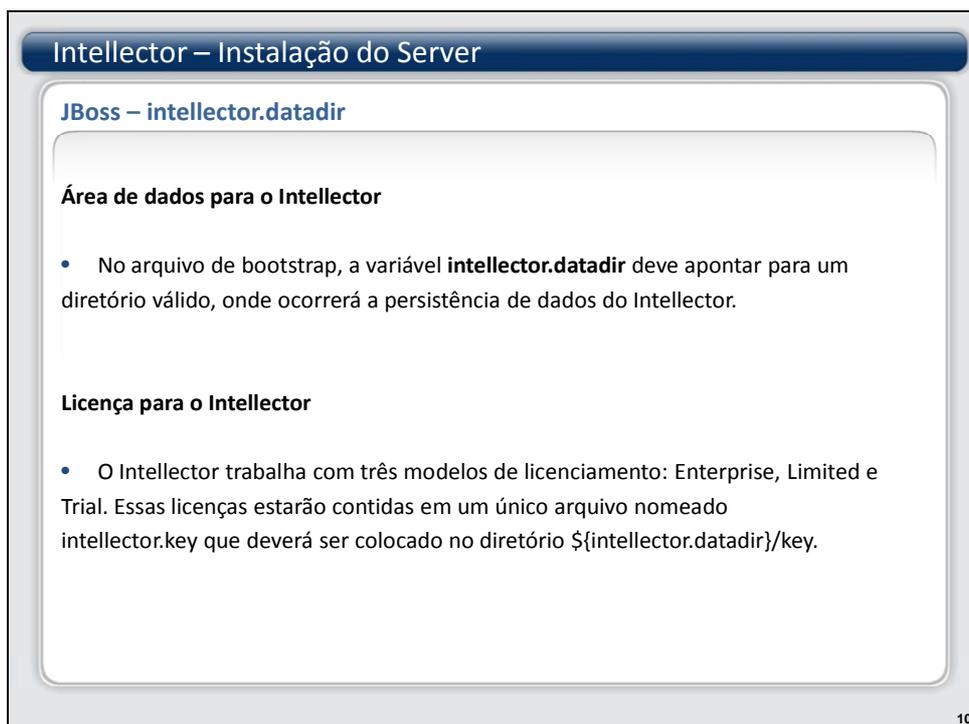
rem inicia o JBoss numa instancia e IP especificos
./run.bat -c default --host 0.0.0.0
```

9

O script de bootstrap para Windows é apresentado no slide.

Para uso no Windows sugerimos usar o nome 'run-intellecator.bat'.

O Script deve ficar em JBOSS\_HOME/bin com permissão explícita de execução.



### **Área de dados para o Intellector**

O Intellector tem uma característica onde, toda a persistência de dados é feita no File System da plataforma.

Esse local de persistência é um diretório ou uma árvore de diretórios.

No arquivo de bootstrap, a variável **intellector.datadir** deve apontar para um diretório válido, onde ocorrerá a persistência e o site criará o restante dos diretórios.

### **Licença para o Intellector**

O Intellector trabalha com três modelos de licenciamento:

- Enterprise - Totalmente livre para execução de qualquer política sem data de expiração.
- Limited - Determinada por um Tipo específico e uma data de expiração ou não.
- Trial - Tipos de políticas livres e com uma data de expiração.

Essas licenças estarão contidas em um único arquivo nomeado **intellector.key** que deverá ser colocado no diretório **Datadir do Intellector**. A licença poderá ser copiada para esse local a qualquer momento, não necessitando de qualquer intervenção no servidor.

## Intellector – Instalação do Server

### JBoss – Deployment do Intellector no JBoss

O próximo passo é o deployment (instalação) do pacote **intellector.war** em uma instância do JBoss. Siga os passos:

- Crie uma pasta **intellector.war** dentro do diretório deploy.
- Extraia o pacote **intellector\_V\_xx\_xxx\_xxx\_xxxxx.war** dentro do diretório recém-criado, **intellector.war** (os X são o controle de versões)

11

### Deployment do Intellector no JBoss

O próximo passo é o deployment do pacote **intellector.war** em uma instância do JBoss. Para isso primeiramente crie uma pasta **intellector.war** dentro do diretório deploy e em seguida, extraia o pacote **intellector.war** dentro do diretório recém-criado. Atenção: Se for uma atualização de versão, simplesmente remova o conteúdo da pasta **intellector.war**, **ANTES** de extrair a nova versão. É boa prática excluir as pastas log, tmp e work, situado abaixo da instância.

## Intellector – Instalação do Server

### JBoss – Iniciando o JBoss

No diretório <JBoss\_HOME>/bin, inicie o *batch*, simplesmente digitando **./run-intellektor.sh** no Linux/Unix ou **run-intellektor.bat** no Windows.

Início do Bootstrap do JBoss:

```
=====
JBoss Bootstrap Environment
JBoss_HOME: /opt/jboss-4.2.3.GA
JAVA: /opt/jdk1.6.0_10/bin/java
JAVA_OPTS: -Dprogram.name=run.sh -server -Xms512m -Xmx536m -XX:PermSize=128m
-XX:MaxPermSize=128m -Xss128k -Dintellektor.datadir=/home/jboss/intellektor-
files -Djava.net.preferIPv4Stack=true
CLASSPATH: /opt/jboss-4.2.3.GA/server/default/lib/log4j.jar:/opt/jboss-
4.2.3.GA/bin/run.jar:/opt/jdk1.6.0_10/lib/tools.jar
=====
17:25:25,676 INFO [Server] Starting JBoss (MX MicroKernel)...
17:25:25,708 INFO [Server] Release ID: JBoss [Trinity] 4.2.3.GA (build:
SVNTag=JBoss_4_2_3_GA date=200807181417)
17:25:25,710 INFO [Server] Home Dir: /opt/jboss-4.2.3.GA
17:25:25,710 INFO [Server] Home URL: file:/opt/jboss-4.2.3.GA/
=====
```

12

### Iniciando o Jboss

Se tudo ocorreu bem até agora, e os passos foram seguidos, iniciar o JBoss é a parte trivial; no diretório JBoss\_HOME/bin, inicie o batch, simplesmente digitando no Linux/Unix ou run-intellektor.bat no Windows.

No slide são exibidos o início e fim do bootstrap do JBoss.

## Intellector – Instalação do Server

### JBoss – Iniciando o JBoss

Fim do Bootstrap do JBoss:

```
17:26:05,961 INFO [intellector] -----
17:26:05,962 INFO [intellector] ----Intellector configurado com sucesso----
17:26:05,962 INFO [intellector] -----
17:26:06,146 INFO [TomcatDeployer] deploy, ctxPath=/jmx-console,
warUrl=../deploy/jmx-console.war/
17:26:07,318 INFO [Server] JBoss (MX MicroKernel) [4.2.3.GA (build:
SVNTag=JBoss_4_2_3_GA date=200807181417)] Started in 21s:602ms
```

13

### Iniciando o Jboss

Se tudo ocorreu bem até agora, e os passos foram seguidos, iniciar o JBoss é a parte trivial; no diretório `JBoss_HOME/bin`, inicie o batch, simplesmente digitando no Linux/Unix ou `run-intellector.bat` no Windows.

No slide são exibidos o início e fim do bootstrap do JBoss.

**Intellector – Instalação do Server**

**JBoss – Configurações adicionais**

**Configurando Pool de Conexões – JNDI**  
Os datasources são configurados em arquivos xxx-ds.xml (onde xxx pode ser qualquer nome) que deverão ser inseridos no diretório deploy do JBoss.

[Data Source para Oracle](#)  
[Data Source para SQL Server](#)

- Também deverão ser colocados no diretório "lib" do JBOSS, os arquivos ".jar" referentes ao driver do banco de dados.
- Driver JNDI para o Oracle :  
[http://www.oracle.com/technology/software/tech/java/sqlj\\_jdbc/htdocs/jdbc\\_10201.html](http://www.oracle.com/technology/software/tech/java/sqlj_jdbc/htdocs/jdbc_10201.html)
- Driver JNDI para o SQLServer :  
[http://sourceforge.net/project/showfiles.php?group\\_id=33291](http://sourceforge.net/project/showfiles.php?group_id=33291)

14

## Configurações adicionais

### Configurando Pool de Conexões – JNDI

Para configurar o Pool de Conexões, insira o Data Source no diretório deploy do JBOSS. O DataSource para Oracle e para SQL Server estão descritos a seguir.

Data Source para Oracle: clique para ver o XML para Oracle.

Data Source para SQL Server: clique para ver o XML para SQL Server.

Também deverão ser colocados no diretório "lib" do JBOSS, os arquivos ".jar" referentes ao driver do banco de dados. Links para download são apresentados no slide.

## Intellector – Instalação do Server

### JBoss – Configurações adicionais – Data Source para Oracle

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- ===== -->
<!-- JBoss Server Configuration -->
<!-- ===== -->
<!-- $Id: oracle-ds.xml 71535 2008-04-01 07:05:03Z adrian@jboss.org $ -->
<!-- ===== -->
<!-- Datasource config for Oracle originally from Steven Coy -->
<!-- ===== -->
<datasources>
  <local-tx-datasource>
    <jndi-name>jdbc/OracleDS</jndi-name>
    <connection-
      url>jdbc:oracle:thin:@192.168.0.149:1521:oralin</connection-url>
    <!-- Here are a couple of the possible OCI configurations.
    For more information, see
    [http://otn.oracle.com/docs/products/oracle9i/doc_library/release
    2/java.920/a96654/toc.htm]
    <connection-url>jdbc:oracle:oci:@youroracle-tns-name</connection-url>
    or
    <connection-
      url>jdbc:oracle:oci:@(description=(address=(host=youroraclehost){
      protocol=tcp}(port=1521))(connect_data={SERVICE_NAME=yourser
      vicename}))</connection-url>
    Clearly, its better to have TNS set up properly.-->
    <driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
    <user-name>cartao_h</user-name>
    <password>uma_senha_qualquer</password>
    <!-- Uses the pingDatabase method to check a connection is still valid
    before handing it out from the pool -->
    <!-- valid-connection-checker-class-
    name>org.jboss.resource.adapter.jdbc.vendor.OracleValidConnectionChe
    cker</valid-connection-checker-class-name-->
    <!-- Checks the Oracle error codes and messages for fatal errors -->
    <exception-sorter-class-
    name>org.jboss.resource.adapter.jdbc.vendor.OracleExceptionSorter</ex
    ception-sorter-class-name>
    <!-- sql to call when connection is created <new-connection-sql>some
    arbitrary sql</new-connection-sql-->
    <!-- sql to call on an existing pooled connection when it is obtained from
    pool - the OracleValidConnectionChecker is preferred<check-valid-
    connection-sql>some arbitrary sql</check-valid-connection-sql-->
    <!-- corresponding type-mapping in the standardjbossmp-jdbc.xml
    (optional)-->
    <metadata>
      <type-mapping>Oracle9i</type-mapping>
    </metadata>
    </local-tx-datasource>
  </datasources>
```

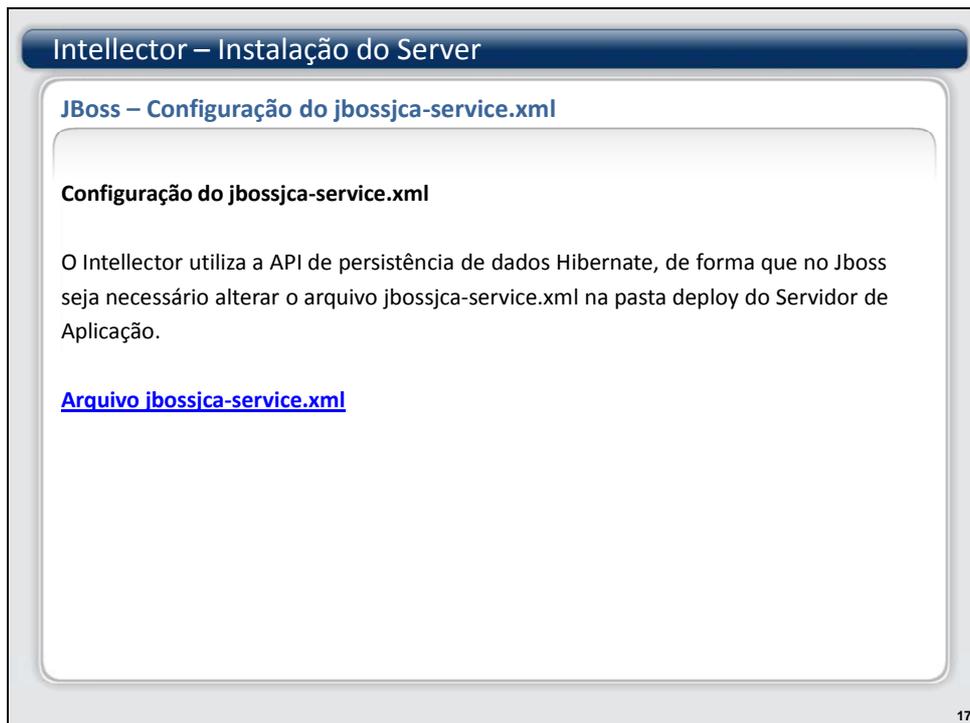
Data Source para Oracle

## Intellecto – Instalação do Server

### JBoss – Configurações adicionais – Data Source para SQL Server

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- ===== -->
<!-- JBoss Server Configuration -->
<!-- ===== -->
<!-- $Id: sqlserver-ds.xml 71535 2008-04-01 07:05:03Z adrian@jboss.org $ -->
<!-- ===== -->
<!-- Datasource config for sqlserver originally from André Riba -->
<!-- ===== -->
<datasources>
  <local-tx-datasource>
    <jndi-name>SqlServerDS</jndi-name>
    <connection-url>jdbc:jtds:sqlserver://192.168.0.85:1433/INDUSTRIAL_D</connection-url>
    <!-- <driver-class>com.microsoft.jdbc.sqlserver.SQLServerDriver</driver-class> -->
    <driver-class>net.sourceforge.jtds.jdbc.Driver</driver-class>
    <user-name>tools</user-name>
    <password>uma_senha_qualquer</password>
    <check-valid-connection-sql>select 1</check-valid-connection-sql>
    <min-pool-size>2</min-pool-size>
    <max-pool-size>10</max-pool-size>
    <!-- corresponding type-mapping in the standardjbosscomp-jdbc.xml (optional) -->
    <metadata>
      <type-mapping>MSSQLSERVER</type-mapping>
    </metadata>
  </local-tx-datasource>
</datasources>
```

Data Source para SQL Server



### **Configuração do jbossjca-service.xml**

O Intellector utiliza a API de persistência de dados Hibernate, de forma que no JBoss seja necessário alterar o arquivo jbossjca-service.xml na pasta deploy do Servidor de Aplicação. Apresentamos no slide as informações que deverão estar presente no arquivo.

Arquivo jbossjca-service.xml: clique para ver o XML.

### JBoss – Configurações adicionais – jbossjca-service.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- ===== -->
<!-- JBoss JCA Configuration -->
<!-- ===== -->
<!-- Sid: jbossjca-service.xml 37786 2005-11-02 20:35:46Z adrian $ -->
<!-- | This contains configuration for the RARDeployer and some xslt based
deployers. -->
<server>
  <!-- JBossCX setup, for J2EE connector architecture support -->
  <!-- The RARDeployer is needed only until xslt based deployment is
written. -->
  <mbean code="org.jboss.util.threadpool.BasicThreadPool"
name="jboss.jca.service=WorkManagerThreadPool">
  <!-- The name that appears in thread names -->
  <attribute name="Name">WorkManager</attribute>
  <!-- The maximum amount of work in the queue -->
  <attribute name="MaximumQueueSize">1024</attribute>
  <!-- The maximum number of active threads -->
  <attribute name="MaximumPoolSize">100</attribute>
  <!-- How long to keep threads alive after their last work (default one
minute) -->
  <attribute name="KeepAliveTime">60000</attribute></mbean>
  <mbean code="org.jboss.resource.work.JBossWorkManager"
name="jboss.jca.service=WorkManager">
  <depends optional-attribute-
name="ThreadPoolName">jboss.jca.service=WorkManagerThreadP
ool</depends>
  <depends optional-attribute-
name="XATerminatorName">jboss.service=TransactionManager</depen
ds></mbean>
  <mbean code="org.jboss.resource.deployment.RARDeployer"
name="jboss.jca.service=RARDeployer">
  <depends optional-attribute-
name="WorkManagerName">jboss.jca.service=WorkManager</depends>
  <depends optional-attribute-
name="XATerminatorName">jboss.service=TransactionManager</depen
ds></mbean>
  <mbean code="org.jboss.deployment.XSLSubDeployer"
name="jboss.jca.service=ConnectionFactoryDeployer">
  <attribute name="DdSuffix">-ds.xml</attribute>
  <attribute name="EnhancedSuffixes">300-ds.xml</attribute>
  <attribute
name="XslUri">stylesheets/ConnectionFactoryTemplate.xsl</attribute>
  <attribute name="ValidateDTDs">>false</attribute></mbean>
  <!-- | The CachedConnectionManager is used partly to relay started
UserTransactions to open connections so they may be enrolled in the new
tx. -->
  <mbean code="org.jboss.resource.connectionmanager.CachedConnection
Manager" name="jboss.jca.service=CachedConnectionManager">
  <depends optional-attribute-
name="TransactionManagerServiceName">jboss.service=TransactionMan
ager</depends>
  <!-- Enable connection close debug monitoring -->
  <!-- <attribute name="Debug">>true</attribute> neste ponto pois a
conexões com o banco de dados não serão gerenciadas pelo Servidor de
Aplicação -->
  <attribute name="Debug">>false</attribute>
</mbean>
</server>

```

jbossjca-service.xml

## Intellector – Instalação do Server

### Intellector-EAR

- O Intellector-EAR é o produto de persistência do Intellector Server.
- Monitora uma fila de dados que são enviados para serem persistidos no banco de dados.
- Para instalar o Intellector-EAR basta colocar o arquivo na pasta deploy do JBoss .

19

### Intellector-EAR

O Intellector-EAR é a produto de persistência do Intellector Server, monitorando uma fila no Servidor de Aplicação onde os dados das execuções de políticas e plugins de acessos são enviados para serem persistidos no banco de dados.

Para instalar o Intellector-EAR basta colocar o arquivo na pasta deploy do JBoss.

## Intellector – Instalação do Server

### Intellector-EAR – Configurando a "fila" do Intellector

- No arquivo **jbossmq-destinations-service.xml** deve ser incluído a fila do Intellector caso o Administrador do Servidor de Aplicação deseje que o Intellector não ocupe uma fila que poderá ser utilizada por outras aplicações.

Neste caso deve-se incluir:

```
<mbean code="org.jboss.mq.server.jmx.Queue"
      name="jboss.mq.destination:service=Queue,name=intellector">
  <depends optional-attribute-
name="DestinationManager">jboss.mq:service=DestinationManager</depends>
</mbean>
```

O arquivo **jbossmq-destinations-service.xml** encontra-se em:

C:\java\jboss-4.2.3.GA\server\default\deploy\jms\jbossmq-destinations-service.xml

20

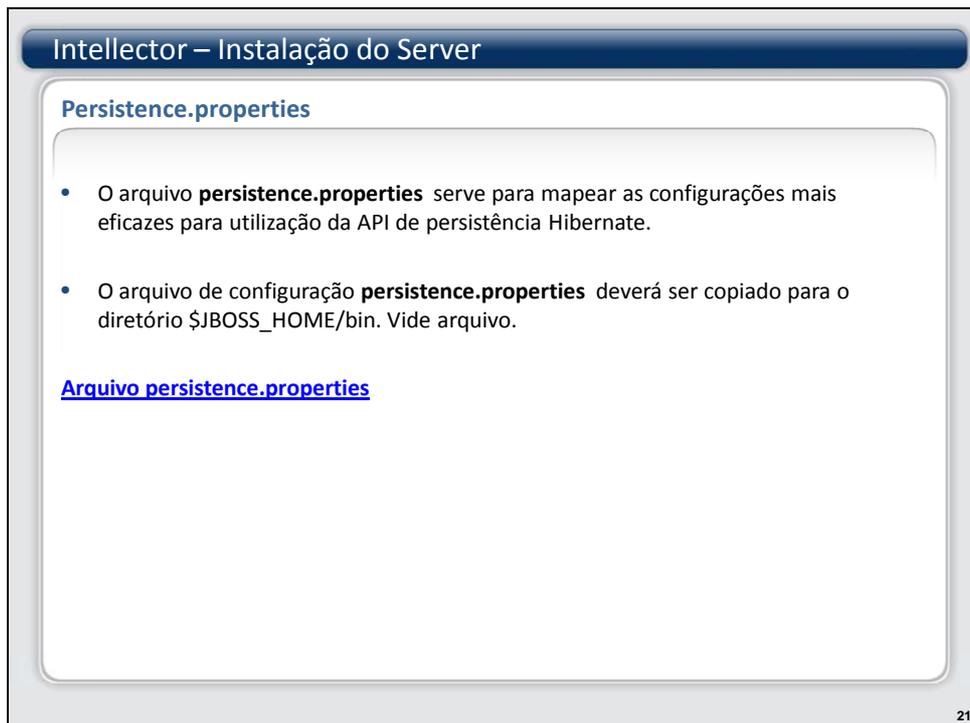
### Configurando a "fila" do Intellector

O arquivo **jbossmq-destinations-service.xml** contém as configurações das filas que o servidor disponibiliza para as aplicações conversarem entre si.

Neste arquivo somente deve ser incluído a fila do Intellector caso o Administrador do Servidor de Aplicação deseje que o Intellector não ocupe uma fila que poderá ser utilizada por outras aplicações.

Neste caso deve-se incluir o código exibido no slide, para que o Intellector possa monitorar uma fila própria.

Para configurar a Fila que o Intellector irá monitorar o Intellector possui um arquivo de configuração para as persistências de dados que serão explicadas mais adiante.



### **Persistence.properties**

Para utilização da persistencia de acessos no 'Intellector Server', o arquivo de configuração **persistence.properties**, deverá ser copiado para o diretório bin do JBoss, como descrito na imagem.

O Intellector ulitiliza a API de persistência Hibernate de forma que este arquivo também serve para mapear as configurações mais eficazes para utilização da API.

Arquivo persistence.properties: clique para ver o arquivo.

## Intellector – Instalação do Server

### Persistence.properties

```
#persistence.properties
#Fri Feb 03 13:52:26 BRST 2012
persistence.queue.name=queue/intellector
persistence.queue.connection.factory=QueueConnectionFactory
persistence.jndi.context.factory=org.jboss.security.jndi.IndiLoginInitialContextFactory
persistence.jndi.address=jnp://127.0.0.1:1099
persistence.jndi.security.principal=
persistence.jndi.security.credentials=
hibernate.connection.driver_class=net.sourceforge.jtds.jdbc.Driver
hibernate.connection.url=jdbc:jtds:sqlserver://192.168.0.242:1433/totvs_d_esquema
#hibernate.default_schema=
hibernate.connection.username=tools
hibernate.connection.password=toolsscc
hibernate.transaction.factory_class=org.hibernate.transaction.JDBCTransactionFactory
hibernate.dialect=org.hibernate.dialect.SQLServerDialect
#hibernate.connection.datasource=java:jdbc/tools
hibernate.jdbc.batch_versioned_data=true
hibernate.cache.use_second_level_cache=false
hibernate.show_sql=true
hibernate.transaction.flush_before_completion=true
hibernate.generate_statistics=true
hibernate.hbm2ddl.auto=update

hibernate.cache.use_query_cache=false
hibernate.max_fetch_depth=1
hibernate.connection.release_mode=after_statement
hibernate.format_sql=true
hibernate.session_factory_name=sessionFactory
hibernate.transaction.auto_close_session=true
hibernate.use_sql_comments=true
hibernate.hbm2ddl.delimiter=;
intellector.usesdatabase=true
hibernate.connection.autocommit=false
hibernate.cache.provider_class=org.hibernate.cache.NoCacheProvider
hibernate.current_session_context_class=thread
```

### Arquivo Persistence.properties

### Conclusão

Neste treinamento, você conheceu um pouco mais sobre:

- Introdução a instalação do

Servidor Intellector

- JDK 6.0
- JBoss
- Intellector-EAR
- Configurando a "fila" do

Intellector

- Persistence.properties



### **Neste treinamento, você conheceu um pouco mais sobre:**

- Introdução a instalação do Servidor Intellector
- JDK 6.0
- JBoss
- Intellector-EAR
- Configurando a "fila" do Intellector
- Persistence.properties

A TOTVS agradece a sua participação.

TOTVS

Este material é de propriedade da TOTVS S>A, sendo proibida a sua reprodução em qualquer meio, total ou parcial, sem aprovação por escrito. Todos os direitos estão reservados. A informação contida aqui é confidencial e não pode ser utilizada fora da empresa ou das franquias que fazem parte da nossa rede, não podendo ser divulgada para clientes, parceiros ou outra empresa ou indivíduo sem o prévio consentimento de um diretor da TOTVS S>A. As opiniões expressas aqui estão sujeitas a modificação sem aviso prévio.

**A TOTVS agradece a sua participação.**