

Versão 10 Release 0
Junho de 2016

Guia PDF de Validação IBM Campaign

IBM

Observação

Antes de usar estas informações e o produto a que elas se referem, leia as informações em “Avisos” na página 17.

Esta edição se aplica à versão 10, liberação 0, modificação 0 do IBM Campaign e a todas as liberações e modificações subsequentes até que seja indicado de outra forma em novas edições.

© Copyright IBM Corporation 1998, 2016.

Índice

Capítulo 1. Visão geral do IBM Validation

Plug-in Developer's Kit (PDK) 1

Configurando o Validation PDK 1

Duas formas de utilizar a API de validação 2

 Construindo um plug-in de classe Java que é carregado no aplicativo 3

 Chamando um aplicativo para manipular a validação 3

Oferta versus validação de campanha 3

Validadores de amostra incluídos no Validation PDK 4

Rotina de teste para o Validation PDK. 4

Scripts de construção para o Validation PDK 5

Capítulo 2. Desenvolvendo plug-ins de validação para o Campaign 7

Configurando o seu ambiente para utilizar o Validation PDK 7

Construindo o validador 8

Configurando o Campaign para usar um plug-in de validação 8

 validationClass 9

 validationClasspath 9

 validatorConfigString 10

Testando a configuração do validador 10

Criando um validador. 11

Cenário de validação de exemplo: Evitar edições de campanha 11

Capítulo 3. Chamando um aplicativo para manipular a validação 13

Configurando o Campaign para utilizar o plug-in executável de amostra 13

Interface de uso executável esperada 13

Antes de contatar o suporte técnico do IBM 15

Avisos 17

Marcas Comerciais 19

Considerações sobre a Política de Privacidade e os

Termos de Uso 19

Capítulo 1. Visão geral do IBM Validation Plug-in Developer's Kit (PDK)

Use o IBM® Validation Plug-in Developer's Kit (PDK) para desenvolver a lógica de validação customizada para uso no IBM Campaign.

É possível criar plug-ins para executar a lógica de validação customizada para campanhas, ofertas, ou ambos.

Alguns usos possíveis de lógica de validação são:

- Para verificação de atributos estendidos (customizados)
- Para fornecer serviços de autorização que estiverem fora do escopo do IBM Marketing Platform (por exemplo, validar quais usuários têm permissão para editar quais atributos estendidos).

O Validation PDK é uma subclasse de uma estrutura de plug-in mais genérica que é fornecida com o IBM Campaign.

O Validation PDK contém informações de referência Javadoc da API do Plug-In e do código de amostra. Para visualizar a documentação, abra o arquivo a seguir em seu navegador da web:

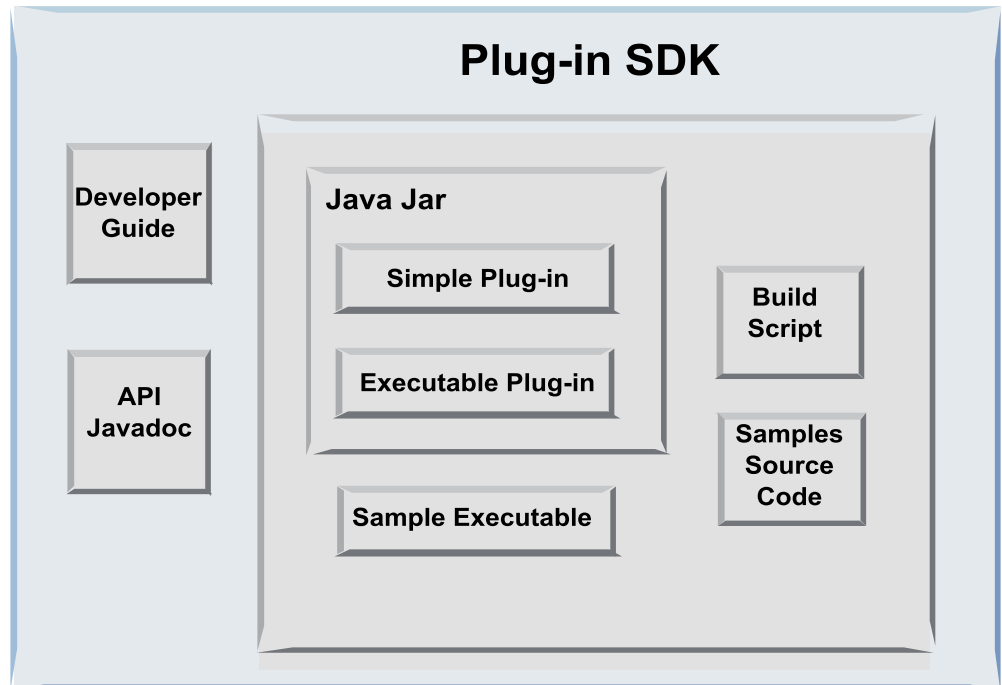
```
C:\IBM\IMS\Campaign_Home\devkits\validation\javadoc\index.html
```

Por exemplo:

```
C:\IBM\IMS\Campaign\devkits\validation\javadoc\index.html
```

Configurando o Validation PDK

O Validation PDK contém componentes para desenvolver plug-ins ou executáveis de linha de comandos Java™ para incluir a validação customizada para o IBM Campaign. O PDK contém exemplos documentados e construíveis de como utilizar o PDK.



A tabela a seguir descreve cada componente.

Tabela 1. Componentes do Validation PDK

Componente	Descrição
Guia do desenvolvedor	Um documento PDF com o título do <i>Guia do IBM Campaign Validation PDK</i> .
API Javadoc	Informações de referência para a API do plug-in.
Arquivo .jar Java	Um arquivo JAR de amostra que contém plug-ins de amostra. O arquivo JAR contém: <ul style="list-style-type: none"> • Plug-in simples: um exemplo de uma classe do validador autocontida. • Plug-in executável: um validador de exemplo que executa um executável de linha de comandos definido pelo usuário para executar a validação.
Executável de amostra	Um executável da linha de comandos que pode ser utilizado com o plug-in executável no UNIX.
Script de Construção	Um script Ant que constrói o código-fonte incluído em plug-ins do validador utilizáveis.
Código-Fonte de Amostra	O código-fonte Java para o validador simples e o validador executável.

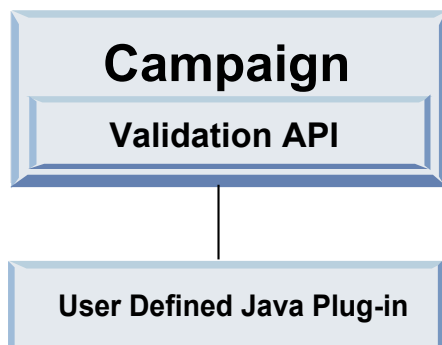
Duas formas de utilizar a API de validação

Há duas maneiras de utilizar a API de validação.

- Utilize para construir um plug-in de classe Java que é carregado no aplicativo.
- Utilize um dos plug-ins incluídos para criar um texto explicativo para um aplicativo executável para manipular a validação.

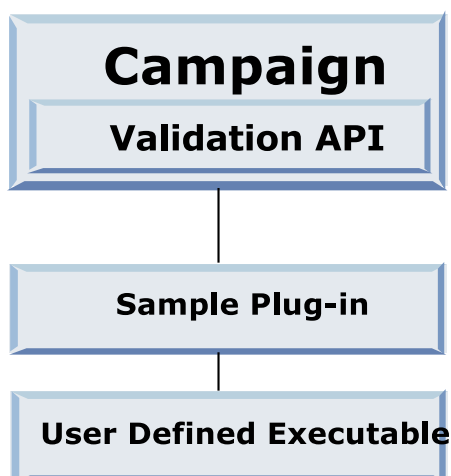
Construindo um plug-in de classe Java que é carregado no aplicativo

O Validation PDK fornece as interfaces, as classes auxiliares e as ferramentas do Developer para desenvolver essas classes.



Chamando um aplicativo para manipular a validação

É possível utilizar um dos plug-ins Validation PDK incluído para criar um texto explicativo para um aplicativo executável para manipular a validação.



O executável pode ser gravado em qualquer linguagem, mas deve residir no servidor IBM Campaign e ser executado no servidor. O plug-in que chama o executável envia em um arquivo XML que contém as informações a serem validadas; por exemplo, o usuário que edita o objeto e os valores antes e depois de todos os atributos padrão e estendidos desse objeto. O IBM Campaign espera informações de resultados na forma de um arquivo XML no retorno.

Oferta versus validação de campanha

Um plug-in que é feito com o Campaign Validation PDK pode executar lógica de validação customizada para campanhas, ofertas, ou ambos.

O PDK Validation pode validar as ofertas e campanhas. Se um plug-in de validação for definido, ele será chamado automaticamente pelo IBM Campaign toda vez que um objeto de oferta ou campanha for salvo. O IBM Campaign

configura um sinalizador ao chamar o método de validação do plug-in. O IBM Campaign transmite os seguintes sinalizadores:

- `ValidationInputData.CAMPAIGN_VALIDATION`, ao incluir ou alterar uma campanha ou
- `ValidationInputData.OFFER_VALIDATION`, quando incluir ou editar uma oferta.

Em seguida, é possível utilizar estes sinalizadores para construir regras de validação para aplicar as ofertas e campanhas.

Validadores de amostra incluídos no Validation PDK

Dois validadores de amostra são incluídos no Campaign Validation PDK: `SimpleCampaignValidator` e `ExecutableCampaignValidator`.

- `SimpleCampaignValidator` é um plug-in autocontido que mostra como realizar ações como autorizações customizadas e validar nomes de campanha permitidos. Ele pode ser localizado no seguinte caminho:

```
devkits\validation\src\com\unica\campaign\core\validation\
samples\SimpleCampaignValidator.Java
```

Recomenda-se fazer uma cópia da classe antes de editá-la para poder reter a versão original, se necessário.

- `ExecutableCampaignValidator` é um plug-in Java que cria um texto explicativo de um aplicativo executável para executar a validação. O código-fonte para o `ExecutableCampaignValidator` é incluído no mesmo diretório que o `SimpleCampaignValidator`:

```
devkits\validation\src\com\unica\campaign\core\validation\
samples\ExecutableCampaignValidator.Java
```

No entanto, o verdadeiro propósito deste exemplo é usar como um executável de linha de comandos para validação. Este arquivo está no seguinte caminho:

```
devkits/validation/src/com/unica/campaign/core/validation/
samples/validate.sh
```

Este arquivo é um executável loopback de amostra que ilustra os tipos comuns de trabalho de validação.

Rotina de teste para o Validation PDK

A capacidade de testar o código de validação sem colocá-lo no IBM Campaign acelera o processo do Developer do plug-in.

Os clientes que utilizam programação extrema e outras metodologias ágeis utilizam extensivamente a unidade de teste. O Validation PDK suporta essas metodologias ao oferecer uma rotina de teste para executar um plug-in fora do Campaign.

Para utilizar a rotina de teste:

1. Altere o caso de teste de unidade para refletir a lógica de validação no plug-in.
2. Execute o script de construção:
 - Para criar o plug-in sem executar nenhum teste de unidade, execute os scripts de construção utilizando o comando "ant jar".
 - Para criar o plug-in e também executar teste de unidade, execute os scripts de construção utilizando o comando "ant run-test".

Scripts de construção para o Validation PDK

Os scripts de construção no PDK Validação compilam todas as classes em um diretório e as coloca em um arquivo JAR que seja adequado para uso no IBM Campaign.

O script de construção fornecido utiliza o seguinte diretório:

```
devkits/validation/src/com/unica/campaign/core/validation/samples/
```

Capítulo 2. Desenvolvendo plug-ins de validação para o Campaign

Um plug-in é uma classe Java que é carregada no tempo de inicialização e chamada sempre que uma campanha ou oferta é validada.

A validação ocorre sempre que um usuário salva uma campanha. É possível criar seus próprios plug-ins Java com as ferramentas que são fornecidas no Validation PDK. O PDK contém o código-fonte para plug-ins de amostra e um arquivo Ant (Apache Ant é uma ferramenta de construção baseada em Java) que é utilizado para compilar plug-ins.

As etapas a seguir explicam como configurar seu ambiente para desenvolver um plug-in e, em seguida, orientam você na criação de seu próprio plug-in.

1. “Configurando o seu ambiente para utilizar o Validation PDK”
2. “Construindo o validador” na página 8
3. “Configurando o Campaign para usar um plug-in de validação” na página 8
4. “Testando a configuração do validador” na página 10
5. “Criando um validador” na página 11

Configurando o seu ambiente para utilizar o Validation PDK

Para utilizar o Validation PDK com o Campaign, deve-se modificar seu caminho e configurar a variável de ambiente `JAVA_HOME`.

O Validation PDK pode ser instalado em qualquer máquina, porém os plug-ins que forem criados com ele deverão estar na máquina na qual o IBM Campaign está em execução. Recomenda-se instalar o PDK na máquina na qual estiver testando seus plug-ins.

O PDK requer que você tenha o Apache Ant e um kit do desenvolvedor Sun Java em sua máquina para criar plug-ins do Java. Para assegurar a compatibilidade, utilize os pacotes Ant e JDK fornecidos com seu servidor de aplicativos.

Para configurar seu ambiente para utilizar o Validation PDK:

1. Inclua a pasta que contém o executável Ant em seu caminho. São fornecidos dois exemplos.
 - Para o WebLogic 11gR1 instalado no diretório padrão no Windows, inclua o seguinte em seu caminho: `C:\Oracle\Middleware\wlserver_10.3\common\bin`
 - Para o WebSphere 7.0 instalado no diretório padrão no Windows, inclua o seguinte em seu caminho: `C:\IBM\WebSphere\AppServer1\bin`
2. Configure a variável de ambiente `JAVA_HOME` para o diretório que contém os diretórios `bin` e `lib` do JDK. São fornecidos dois exemplos.
 - Para o WebLogic 11gR1 no Windows, configure `JAVA_HOME` para `C:\Oracle\Middleware\jdk160_18`
 - Para o WebSphere 7.0 no Windows, configure `JAVA_HOME` para `C:\IBM\WebSphere\AppServer1\java\jre`

Construindo o validador

O IBM Campaign Validation PDK fornece um script Ant que pode construir todo o código nos arquivos de amostra.

O comportamento padrão para o script é criar um jar que contém as classes de validação. Opcionalmente, ele também pode criar Javadoc e executar testes com relação aos validadores para assegurar que eles funcionem no Campaign antes de tentar utilizar o plug-in em produção.

Para construir o validador:

1. Mude para o diretório PDK `<IBM_IMS_Home\Campaign_Home>\devkits\validation\build`

Por exemplo: `C:\IBM\IMS\Campaign\devkits\validation\build`

Esse diretório contém o script Ant, `build.xml`.

2. Execute o jar Ant na linha de comandos.
 - Para criar o plug-in sem fazer nenhum teste de unidade, use o comando `ant jar`.
 - Para criar o plug-in e também fazer testes de unidade, use o comando `ant run-test`.

O Ant executa o script e produz um arquivo JAR denominado `validator.jar` no subdiretório `lib`. Por exemplo:

`C:\IBM\IMS\Campaign\devkits\validation\build\lib`

Agora você tem um validador customizado que pode ser utilizado no IBM Campaign. A próxima etapa é configurar o Campaign para usar esse validador.

Configurando o Campaign para usar um plug-in de validação

Para configurar o Campaign para usar um plug-in de validação, utilize as definições de configuração em `Campaign > partitions > partition[n] > validation`.

As propriedades de configuração informam ao Campaign como localizar a classe de plug-in fornecem uma maneira de transmitir as informações de configuração para os plug-ins.

Nota: A validação funciona com diversas partições; `partition[n]` pode ser alterada para qualquer nome de partição para fornecer rotinas de validação também para essas partições.

É possível ajustar as seguintes definições de configuração de validação:

- “`validationClass`” na página 9
- “`validationClasspath`” na página 9
- “`validatorConfigString`” na página 10

Para usar o `SimpleCampaignValidator`, configure as propriedades conforme a seguir:

- `validationClasspath`: `Unica\campaign\devkits\validation\lib\validator.jar`
- `validationClass`:
`com.unica.campaign.core.validation.samples.SimpleCampaignValidator`

- O `validatorConfigString` não precisa ser configurado para usar `SimpleCampaignValidator` porque ele não usa uma sequência de configuração.

Para usar o `ExecutableCampaignValidator`, configure as propriedades conforme a seguir:

- `validationClasspath`: `<Campaign_home>\devkits\validation\lib\validator.jar`
- `validationClass`:
`com.unica.campaign.core.validation.samples.ExecutableCampaignValidator`
- O `validatorConfigString`: `<Campaign_home>\pdk\bin\validate.sh`

validationClass

O `validationClass` informa ao Campaign o nome da classe a ser utilizada para validação com um Validation PDK.

Propriedade	Descrição
Descrição	O nome da classe a ser utilizada para validação. O valor da propriedade <code>validationClasspath</code> indica a localização desta classe.
Detalhes	A classe deve ser completamente qualificada com seu nome de pacote. Se esta característica não estiver definida, o Campaign não faz nenhuma validação customizada.
Exemplo	<code>com.unica.campaign.core.validation.samples.SimpleCampaignValidator</code> Esse exemplo define <code>validationClass</code> para a classe <code>SimpleCampaignValidator</code> a partir do código de amostra.
Padrão	Por padrão, nenhum caminho está configurado: <code><property name="validationClass" /></code>

validationClasspath

O `validationClasspath` informa ao Campaign o local da classe a ser utilizada para validação com um plug-in Validation PDK.

Propriedade	Descrição
Descrição	O caminho para a classe que é utilizada para validação customizada.
Detalhes	Use um caminho completo ou caminho relativo. Se o caminho for relativo, o comportamento dependerá do servidor de aplicativos que estiver executando o Campaign. O WebLogic usa o caminho para o diretório de trabalho do domínio que, por padrão, é <code>c:\bea\user_projects\domains\mydomain.</code> Se o caminho terminar com uma barra (/ para UNIX ou \ para Windows), o Campaign assumirá que ele aponta para o local da classe de plug-in Java a ser utilizada. Se o caminho não terminar com uma barra, o Campaign assumirá que ele é o nome de um arquivo .jar que contém a classe Java, conforme mostrado no exemplo a seguir. Se o caminho não for configurado, o Campaign não tentará carregar um plug-in.

Propriedade	Descrição
Exemplo	<code>/<CAMPAIGN_HOME>/devkits/validation/lib/validator.jar</code> Este é o caminho em uma plataforma UNIX que aponta para o arquivo JAR que é fornecido com o kit do desenvolvedor do plug-in.
Padrão	Por padrão, nenhum caminho está configurado: <code><property name="validationClasspath" /></code>
Consulte também	Consulte "validationClass" na página 9 para obter informações sobre como designar a classe a ser utilizada.

validatorConfigString

O `validatorConfigString` é transmitido para o plug-in validador quando ele é carregado pelo Campaign.

Propriedade	Descrição
Descrição	Uma sequência que é transmitida para o plug-in do validador quando ele é carregado pelo Campaign.
Detalhes	O designer determina como o plug-in utiliza essa sequência. É possível usá-lo para enviar uma sequência de configuração para seu plug-in quando o sistema carregá-lo. Por exemplo, o <code>ExecutableCampaignValidator</code> (a partir do plug-in executável de amostra incluído no PDK) usa essa propriedade para indicar o executável a ser executado.
Exemplo	Para executar o script de shell Bourne de amostra como o script de validação, configure <code>validatorConfigString</code> para <code>/opt/unica/campaign/devkits/validation/src/com/unica/campaign/core/validation/samples/validate.sh</code>
Padrão	Por padrão, nenhum caminho está configurado: <code><property name="validatorConfigString" /></code>

Testando a configuração do validador

Depois de construir o arquivo `struts` que contém a classe `SimpleCampaignValidator` e fazer as mudanças na configuração necessárias, é possível testar e utilizar o plug-in.

O exemplo de plug-in a seguir impede que os usuários do Campaign salvem uma campanha denominada "badCampaign."

Para testar sua configuração:

1. Reimplante seu servidor de aplicativos para que as mudanças entrem em vigor. Para obter instruções, consulte a documentação do seu servidor.
2. Efetue login no IBM Campaign e acesse a página de criação da campanha.
3. Crie uma campanha com o nome **badCampaign** e tente salvar.

Se tudo estiver configurado corretamente, você não conseguirá salvar a nova campanha. Se uma mensagem de erro do validador for recebida, você saberá que a configuração está funcionando corretamente.

Criando um validador

Siga estas instruções para criar um plug-in de validação muito semelhante ao `SimpleCampaignValidator`, mas impede a criação de campanhas que sejam denominadas "badCampaign2."

1. Faça uma cópia do validador de amostra `SimpleCampaignValidator.java` em `<IBM_IMS_Home\Campaign_Home>\devkits\validation\src\com\unica\campaign\core\validation\samples`. Nomeie a cópia do `MyCampaignValidator.java` e deixe-a no mesmo diretório que a origem. Por exemplo:
`C:\IBM\IMS\Campaign\devkits\validation\src\com\unica\campaign\core\validation\samples\MyCampaignValidator.java`
2. Abra `MyCampaignValidator.java` em um editor. Localize a palavra "badCampaign" no documento e substitua-a pela palavra "badCampaign2."
3. Salve o arquivo e feche o editor.
4. Construa os validadores novamente. Para obter detalhes, consulte "Construindo o validador" na página 8. Se seu servidor de aplicativos bloquear o arquivo `validate.jar` enquanto em uso, pare o servidor antes de construir os validadores.
5. Reconfigure `campaign_config.xml` para usar a nova classe: `<property name="validationClass" value="com.unica.campaign.core.validation.samples.MyCampaignValidator">`
6. Teste o validador. Para obter detalhes, consulte "Testando a configuração do validador" na página 10.

Confirme se o validador funciona: você não deve ser capaz de salvar campanhas denominadas "badCampaign2."

Cenário de validação de exemplo: Evitar edições de campanha

Este exemplo explica como usar a validação para evitar edições específicas de uma campanha.

Se desejar impedir que alguém que está editando uma campanha altere o código da campanha, é possível utilizar uma rotina de validação de campanha customizada. Essa rotina assegura que a verificação a seguir seja feita quando a campanha for salva:

```
new_campaign_code == old_campaign_code
```

Para manipular o caso quando a campanha for criada pela primeira vez, transmita à rotina um sinalizador que indica se a campanha que está sendo validada é nova (criação) ou existente (edição). Se esse sinalizador indicar **edição**, compare os códigos da campanha.

O aplicativo Campaign configura este sinalizador no objeto `InputValidationData` que é, então, transmitido para o plug-in. O plug-in lê o sinalizador quando ele determinar se a validação é para uma campanha nova ou alterada.

Capítulo 3. Chamando um aplicativo para manipular a validação

O Validation PDK inclui um validador de amostra ExecutableCampaignValidator que executa um executável `validate.sh` a partir da linha de comandos para executar a validação.

As seguintes seções explicam como:

- Configurar o Campaign para executar o plug-in executável de amostra e;
- Criar seu próprio plug-in executável que esteja em conformidade com a utilização da interface de uso executável.

Configurando o Campaign para utilizar o plug-in executável de amostra

Para usar o ExecutableCampaignValidator, ajuste as definições de configuração em Campaign > partitions > partition[n] > validation.

Configure as propriedades da seguinte forma:

- validationClasspath:
`<Campaign_home>\devkits\validation\lib\validator.jar`
- validationClass:
`com.unica.campaign.core.validation.samples.ExecutableCampaignValidator`
- validatorConfigString:
`<Campaign_home>\pdk\bin\validate.sh`

O script de amostra que é fornecido com a Validação PDK é um script de shell Bourne para UNIX. Ele nega a criação da campanha para qualquer pessoa que tiver o nome do usuário "badUser." É possível visualizar o código para esse executável no seguinte diretório:

```
devkits\validation\src\com\unica\campaign\core\validation\
samples\validate.sh
```

É necessário desenvolver seu próprio script que executa uma validação relevante de sua implementação. Linguagens de script, como PERL e Python, são boas candidatas para scripts de processamento de texto como este; no entanto, qualquer linguagem que possa ser executada a partir da linha de comandos é aceitável.

Interface de uso executável esperada

O plug-in ExecutableCampaignValidator chama um arquivo executável com uma linha de comandos que contém os argumentos a seguir.

- executable_name: A sequência configurada no validatorConfigString no IBM Marketing Platform.
- data_filename: O nome do arquivo que o executável lê como entrada. Os dados de entrada devem ser formatados em XML.
- expected_result_filename: O nome do arquivo que o executável deverá enviar como saída. Os resultados esperados estão no formato data XXX.xml, em que XXX é um número.

- A seguir há um exemplo de como os dados com sucesso são enviados:
`<ValidationResult result="0" generalFailureMessage="" />`
- A seguir há um exemplo de como os dados com falha são enviados:
`<ValidationResult result="1" generalFailureMessage="">
 <AttributeError attributeName="someAttribute" errorMessage="something" />
 <AttributeError attributeName="someAttribute2" errorMessage="something2" />
</ValidationResult>`
- O texto no arquivo XML deve ser codificado em caracteres ASCII ou UTF-8 regulares.

Nota: É altamente recomendado fornecer aos usuários mensagens de erro fáceis de entender para que eles possam corrigir o problema antes de tentarem outra operação de salvamento.

Antes de contatar o suporte técnico do IBM

Se você encontrar um problema que não pode resolver consultando a documentação, o contato responsável por suporte de sua empresa poderá registrar uma chamada no suporte técnico da IBM. Siga estas diretrizes para assegurar que seu problema seja resolvido de modo eficiente e bem-sucedido.

Se você não possuir um contato responsável por suporte em sua empresa, entre em contato com o administrador do IBM para obter mais informações.

Nota: O Suporte Técnico não grava ou cria scripts de API. Para obter assistência para implementar nossas ofertas de API, entre em contato com os Serviços profissionais da IBM.

Informações para reunir

Antes de contatar o suporte técnico IBM, reúna as informações a seguir:

- Uma breve descrição da natureza de seu problema.
- Mensagens de erro detalhadas que você vê quando ocorre o problema.
- Etapas detalhadas para reproduzir o problema.
- Arquivos de log relacionados, arquivos de sessão, arquivos de configuração e arquivos de dados.
- Informações sobre seu produto e ambiente do sistema , que você pode obter conforme descrito em "Informações do sistema".

Informações sobre o sistema

Ao ligar para o suporte técnico da IBM, pode ser solicitado que forneça informações sobre o seu ambiente.

Se seu problema não impedir que você efetue login, muitas destas informações estão disponíveis na página Sobre, que fornece informações sobre seus aplicativos do IBM instalados.

É possível acessar a página Sobre selecionando **Ajuda > Sobre**. Se a página Sobre não estiver acessível, verifique o arquivo `version.txt` que se localiza no diretório de instalação do aplicativo.

Informações de contato para o suporte técnico IBM

Para obter maneiras de contatar o suporte técnico IBM, consulte o website de Suporte Técnico do Produto IBM: (http://www.ibm.com/support/entry/portal/open_service_request).

Nota: Para inserir uma solicitação de suporte, deve-se efetuar login em uma conta do IBM. Essa conta deve estar vinculada ao seu número de cliente IBM. Para saber mais sobre como associar sua conta ao seu número de cliente do IBM, consulte **Recursos de suporte > Suporte de software autorizado** no Portal de suporte.

Avisos

Estas informações foram desenvolvidas para produtos e serviços oferecidos nos Estados Unidos.

É possível que a IBM não ofereça os produtos, serviços ou recursos discutidos nesta publicação em outros países. Consulte seu representante IBM local para obter informações sobre os produtos e serviços atualmente disponíveis em sua área. Qualquer referência a produtos, programas ou serviços IBM não significa que apenas produtos, programas ou serviços IBM possam ser utilizados. Qualquer produto, programa ou serviço funcionalmente equivalente, que não infrinja nenhum direito de propriedade intelectual da IBM ou quaisquer outros direitos da IBM poderá ser utilizado em substituição a este produto, programa ou serviço. Entretanto, é responsabilidade do usuário avaliar e verificar a operação de qualquer produto, programa ou serviço não IBM.

A IBM pode ter patentes ou solicitações de patentes pendentes relativas a assuntos descritos nesta publicação. O fornecimento desta publicação não lhe garante direito algum sobre tais patentes. Pedidos de licenças devem ser enviados, por escrito, para:

Gerência de Relações Comerciais e Industriais da IBM Brasil
Av. Pasteur, 138-146
Botafogo
Rio de Janeiro
CEP: 22290-240

Para pedidos de licenças com relação a informações sobre DBCS (Conjunto de Caracteres de Byte Duplo), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie pedidos, por escrito, para:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

O parágrafo a seguir não se aplica a nenhum país em que tais disposições não estejam de acordo com a legislação local: A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO "NO ESTADO EM QUE SE ENCONTRA", SEM GARANTIA DE NENHUM TIPO, SEJA EXPRESSA OU IMPLÍCITA, INCLUINDO, MAS A ELAS NÃO SE LIMITANDO, AS GARANTIAS IMPLÍCITAS DE NÃO INFRAÇÃO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM DETERMINADO PROPÓSITO. Alguns países não permitem a exclusão de garantias expressas ou implícitas em certas transações; portanto, essa disposição pode não se aplicar ao Cliente.

Essas informações podem conter imprecisões técnicas ou erros tipográficos. São feitas alterações periódicas nas informações aqui contidas; tais alterações serão incorporadas em futuras edições da publicação. A IBM pode, a qualquer momento, aperfeiçoar e/ou alterar os produtos e/ou programas descritos nesta publicação, sem aviso prévio.

Referências nestas informações a Web sites não-IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a estes Web sites. Os materiais contidos nesses Web sites não fazem partes dos materiais deste produto IBM e a utilização desses Web sites é de inteira responsabilidade do Cliente.

A IBM pode utilizar ou distribuir as informações fornecidas da forma que julgar apropriada sem incorrer em qualquer obrigação para com o Cliente.

Licenciados deste programa que desejam obter informações sobre este assunto com objetivo de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) a utilização mútua das informações trocadas, devem entrar em contato com:

Gerência de Relações Comerciais e Industriais da IBM Brasil
Av. Pasteur, 138-146
Botafogo
Rio de Janeiro
CEP: 22290-240

Tais informações podem estar disponíveis, sujeitas a termos e condições apropriados, incluindo em alguns casos, o pagamento de uma taxa.

O programa licenciado descrito nesta publicação e todo o material licenciado disponível são fornecidos pela IBM sob os termos do Contrato com o Cliente IBM, do Contrato Internacional de Licença do Programa IBM ou de qualquer outro contrato equivalente.

Todos os dados de desempenho aqui contidos foram determinados em um ambiente controlado. Portanto, os resultados obtidos em outros ambientes operacionais podem variar significativamente. Algumas medidas podem ter sido tomadas em sistemas em nível de desenvolvimento e não há garantia de que tais medidas sejam iguais em sistemas geralmente disponíveis. Além disso, algumas medidas podem ter sido estimadas por extrapolação. Os resultados reais podem ser diferentes. Os usuários desta publicação devem verificar os dados aplicáveis para seu ambiente específico.

As informações referentes a produtos não IBM foram obtidas junto aos fornecedores desses produtos, anúncios publicados ou outras fontes de publicidade disponíveis. A IBM não testou aqueles produtos e não pode confirmar a precisão do desempenho, compatibilidade ou qualquer outra reivindicação relacionada a produtos que não sejam da IBM. Dúvidas sobre os recursos de produtos não IBM devem ser encaminhadas diretamente a seus fornecedores.

Todas as declarações relacionadas aos objetivos e intenções futuras da IBM estão sujeitas a mudanças ou cancelamento sem aviso prévio e representam apenas metas e objetivos.

Todos os preços da IBM mostrados, são preços de varejo sugeridos pela IBM, são atuais e estão sujeitos à mudança sem aviso. Os preços para o revendedor podem variar.

Essas informações contêm exemplos de dados e relatórios utilizados em operações diárias de negócios. Para ilustrá-los da forma mais completa possível, os exemplos

incluem nomes de pessoas, empresas, marcas e produtos. Todos estes nomes são fictícios e qualquer semelhança com nomes e endereços utilizados por uma empresa real é mera coincidência.

LICENÇA DE COPYRIGHT:

Estas informações contêm programas aplicativos de amostra no idioma de origem, que ilustram técnicas de programação em diversas plataformas operacionais. O Cliente pode copiar, modificar e distribuir estes programas de amostra de qualquer forma, sem a necessidade de pagar a IBM, para os propósitos de desenvolvimento, uso, marketing ou distribuição de programas aplicativos em conformidade com a interface de programação de aplicativos para a plataforma operacional para a qual os programas de amostra são escritos. Esses exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar confiabilidade, manutenção ou função destes programas. Os programas de amostra são fornecidos "NO ESTADO EM QUE SE ENCONTRAM", sem garantia de nenhum tipo. A IBM não se responsabilizará por nenhum dano derivado do uso dos programas de exemplo.

Se estas informações estiverem sendo exibidas em cópia eletrônica, as fotografias e ilustrações coloridas podem não aparecer.

Marcas Comerciais

IBM, o logotipo IBM e ibm.com são marcas ou marcas comerciais da International Business Machines Corp., registradas em muitos países no mundo todo. Outros nomes de serviços e produtos podem ser marcas comerciais da IBM ou de outras empresas. Uma lista atual de marcas comerciais da IBM está disponível na Web em "Copyright and trademark information" em www.ibm.com/legal/copytrade.shtml.

Considerações sobre a Política de Privacidade e os Termos de Uso

Produtos de Software IBM, incluindo soluções de software como serviço, ("Ofertas de Software") podem usar cookies ou outras tecnologias para coletar informações de uso do produto, para ajudar a melhorar a experiência do usuário final, para customizar interações com o usuário final ou para outros propósitos. Um cookie é uma parte dos dados que o website pode enviar para o seu navegador, e que pode ser armazenada em seu computador como uma tag que identifica o seu computador. Em muitos casos, nenhuma informação pessoal é coletada por esses cookies. Se uma Oferta de Software que você está usando permitir a você coletar informações pessoais por meio de cookies e tecnologias semelhantes, informamos a você sobre as especificações abaixo.

Dependendo das configurações implementadas, esta Oferta de Software pode usar cookies de sessão e persistentes que coletam o nome de usuário de cada usuário e outras informações pessoais com propósitos de gerenciamento de sessão, usabilidade de usuário aprimorada ou outros propósitos de rastreamento de uso ou funcionais. Esses cookies podem ser desativados, mas a desativação também eliminará a funcionalidade ativada por eles.

Várias jurisdições regulamentam a coleta de informações pessoais por meio de cookies e tecnologias semelhantes. Se as configurações implementadas para esta Oferta de Software fornecer ao Cliente a capacidade de coletar informações pessoais dos usuários finais usando cookies e outras tecnologias, você deverá procurar o aviso legal sobre qualquer lei aplicável a essa coleta de dados, incluindo todos os requisitos para fornecer avisos e consentimento quando adequado.

A IBM requer que os Clientes (1) forneçam um link claro e evidente para os termos de uso do website do Cliente (por exemplo, política de privacidade) que inclui um link para a coleção de dados da IBM e do Cliente e práticas de uso, (2) notifiquem que cookies e gifs/web beacons claros estão sendo colocados no computador do visitante pela IBM em nome do Cliente juntamente com uma explicação do propósito de tal tecnologia, e (3) até a extensão requerida por lei, obtenham o consentimento dos visitantes do website antes de se colocar cookies e gifs/web beacons claros pelo Client ou IBM em nome do Cliente nos dispositivos do visitante do website

Para obter mais informações sobre o uso de várias tecnologias, incluindo cookies, para estes propósitos, consulte Declaração de Privacidade Online da IBM em: <http://www.ibm.com/privacy/details/us/en> na seção intitulada "Cookies, Web Beacons e Outras Tecnologias".



Impresso no Brasil