

IBM Campaign
Versão 9 Release 1.1
26 de novembro de 2014

Guia PDK de Validação

IBM

Nota

Antes de usar estas informações e o produto suportado por elas, leia as informações em “Avisos” na página 15.

Esta edição se aplica à versão 9, liberação 1, modificação 1 de IBM Campaign e a todas as liberações e modificações subsequentes até que seja indicado de maneira diferente em novas edições.

© Copyright IBM Corporation 1998, 2014.

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	2
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	4

Capítulo 2. Desenvolvendo plug-ins de validação para o Campaign 5

Configurando o seu ambiente para utilizar o Validation PDK	5
Construindo o validador	6
Configurando o Campaign para usar um plug-in de validação	6

validationClass	7
validationClasspath	7
validatorConfigString	7
Testando a configuração do validador	8
Criando um validador	8
Cenário de validação de exemplo: Evitar edições de campanha	9

Capítulo 3. Chamando um aplicativo para manipular a validação 11

Configurando o Campaign para utilizar o plug-in executável de amostra	11
Interface de uso executável esperada	11

Entrando em Contato com o Suporte Técnico do IBM 13

Avisos 15

Marcas Registradas	17
------------------------------	----

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 seguinte arquivo em qualquer navegador da Web:

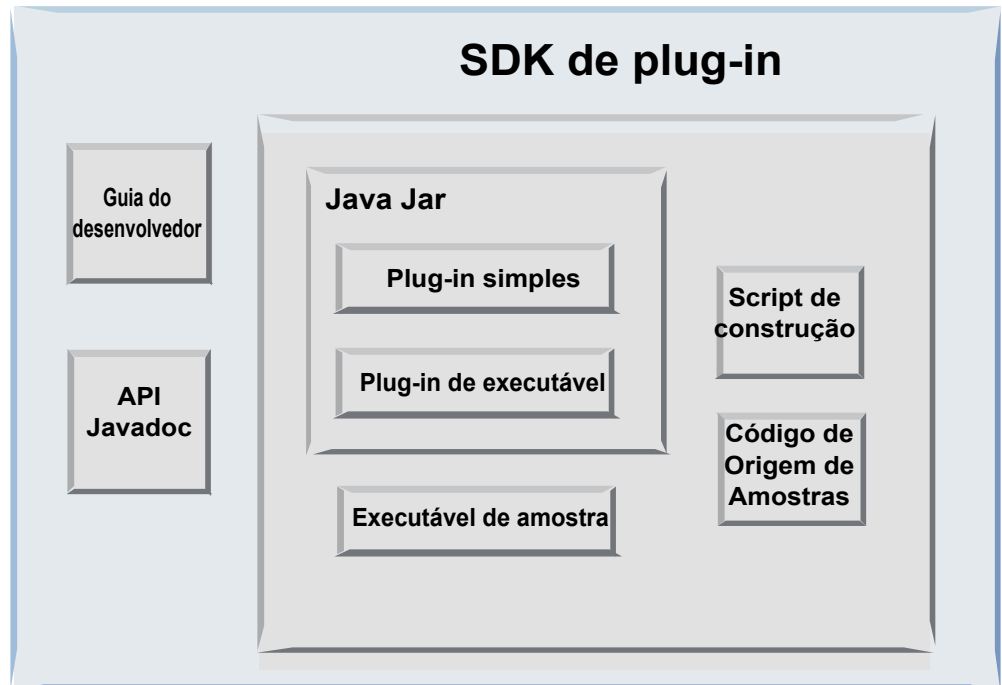
`C:\IBM_EMM_Home\Campaign_Home\devkits\validation\javadoc\index.html`

Por exemplo:

`C:\IBM\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.

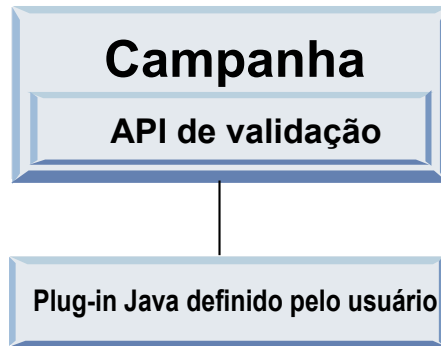
Duas formas de utilizar a API de validação

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.

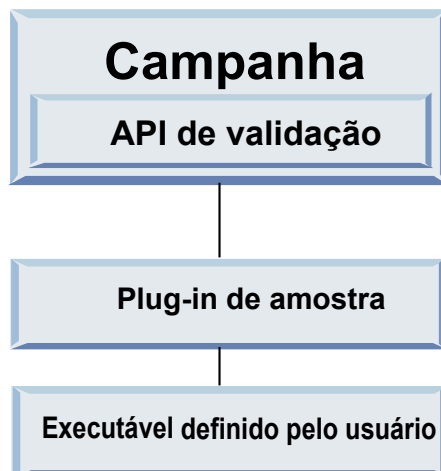
Construindo um plug-in de classe Java que é carregado no aplicativo

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 6
3. “Configurando o Campaign para usar um plug-in de validação” na página 6
4. “Testando a configuração do validador” na página 8
5. “Criando um validador” na página 8

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 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 o diretório para o diretório PDK,

```
<IBM_EMM_Home\Campaign_Home>\devkits\validation\build
```

O script Ant, o `build.xml`, é exibido neste diretório.

2. Execute o jar Ant na linha de comandos.

- Para criar o plug-in sem fazer nenhum teste de unidade, utilize o comando "ant jar".
- Para criar o plug-in e também fazer testes de unidade, utilize o comando "ant run-test".

O Ant executa o script e produz um arquivo JAR denominado `legacy.jar` no diretório:

```
<IBM_EMM_Home\Campaign_Home>\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 7
- "validationClasspath" na página 7
- "validatorConfigString" na página 7

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 validationClasspath 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	com.unica.campaign.core.validation.samples.SimpleCampaignValidator Esse exemplo define validationClass para a classe SimpleCampaignValidator a partir do código de amostra.
Padrão	Por padrão, nenhum caminho está configurado: <property name="validationClass" />

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 c:\bea\user_projects\domains\mydomain. 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.
Exemplo	/<CAMPAIGN_HOME>/devkits/validation/lib/validator.jar 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: <property name="validationClasspath" />
Consulte também	Consulte "validationClass" 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 ExecutableCampaignValidator (a partir do plug-in executável de amostra incluído no PDK) usa essa propriedade para indicar o executável a ser executado.

Propriedade	Descrição
Exemplo	Para executar o script de shell Bourne de amostra como o script de validação, configure validatorConfigString para /opt/unica/campaign/devkits/validation/src/com/unica/campaign /core/validation/samples/validate.sh
Padrão	Por padrão, nenhum caminho está configurado: <property name="validatorConfigString" />

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

Essas instruções explicam como criar um plug-in de validação muito semelhante ao SimpleCampaignValidator, mas que impede a criação de campanhas que sejam denominadas "badCampaign2."

Para criar um validador:

1. Faça uma cópia do validador de amostra do SimpleCampaignValidator.java, localizado em

```
<IBM_EMM_Home\Campaign_Home>\devkits\validation\src\com\
unica\campaign\core\validation\samples
```
2. Nomeie a cópia do MyCampaignValidator.java e deixe-a no mesmo diretório que a origem.
3. Abra MyCampaignValidator.java em um editor. Localize a palavra "badCampaign" no documento e substitua-a pela palavra "badCampaign2."
4. Salve o arquivo e feche o editor.
5. Construa os validadores novamente. Para obter detalhes, consulte "Construindo o validador" na página 6.

Nota: Se seu servidor de aplicativos bloquear o arquivo validate.jar enquanto estiver em uso, deve-se parar o servidor antes de construir os validadores.

6. Reconfigure campaign_config.xml para utilizar a nova classe:

```
<property name="validationClass"
value="com.unica.campaign.core.validation.samples.MyCampaignValidator">
```

7. Teste o validador. Para obter detalhes, consulte "Testando a configuração do validador".

Provavelmente você não conseguirá 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.

Entrando em Contato com o Suporte Técnico do IBM

Se encontrar um problema que não puder resolver consultando a documentação, o contato de suporte designado por sua empresa pode registrar uma chamada com o suporte técnico da IBM . Use as informações nesta seção para assegurar que seu problema seja eficientemente resolvido com êxito.

Se você não for um contato de suporte designado em sua empresa, entre em contato com seu administrador da IBM para obter informações.

Informações a Serem Reunidas

Antes de entrar em contato com o suporte técnico da IBM , reúna as informações a seguir:

- Uma breve descrição da natureza de seu problema.
- Mensagens de erro detalhadas que você vê quando o problema ocorre.
- Etapas detalhadas para reproduzir o problema.
- Arquivos de log relacionados, arquivos de sessão, arquivos de configuração e arquivos de dados.
- As informações sobre seu ambiente de produto e sistema, que podem ser obtidas como descrito em "Informações de sistema".

Informações de Sistema

Ao ligar para o suporte técnico da IBM , pode ser que você seja solicitado a fornecer informações sobre seu ambiente.

Se o seu problema não impedi-lo de efetuar login, a maior parte dessas informações está disponível na página Sobre, que fornece as informações sobre seus aplicativos IBM instalados.

É possível acessar a página Sobre selecionando **Ajuda > Sobre**. Se a página Sobre não estiver acessível, é possível obter o número da versão de qualquer aplicativo IBM visualizando o arquivo `version.txt` localizado sob o diretório de instalação para cada aplicativo.

Informações de Contato para o Suporte Técnico da IBM

Para obter as formas de contato com o suporte técnico da IBM , consulte o website do Suporte Técnica do Produto IBM : (<http://www.unica.com/about/product-technical-support.htm>).

Avisos

Estas informações foram desenvolvidas para produtos e serviços oferecidos nos E.U.A.

O IBM pode não oferecer os produtos, serviços ou recursos discutidos neste documento em outros países. Consulte seu representante IBM local para obter informações sobre os produtos e serviços atualmente disponíveis em sua área. Toda referência a um produto, programa ou serviço IBM não tem a intenção de declarar ou implica em que apenas esse produto, programa ou serviço IBM possa ser usado. Qualquer produto, programa ou serviço funcionalmente equivalente que não infrinja nenhum direito de propriedade intelectual da IBM pode ser usado alternativamente. Todavia, 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 que cobrem os assuntos descritos neste documento. O fornecimento deste documento não concede ao Cliente nenhuma licença a essas patentes. Pedidos de licença podem ser enviados, por escrito, para:

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

Para perguntas sobre licença relacionadas a informações de byte duplo (DBCS), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie dúvidas, por escrito ao:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

O parágrafo a seguir não se aplica ao Reino Unido ou qualquer outro 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 estados não permitem a exclusão de garantias expressas ou implícitas em determinadas transações, portanto, essa declaração pode não se aplicar ao Cliente.

Estas informações podem incluir imprecisões técnicas ou erros tipográficos. Alterações são periodicamente feitas nas informações aqui existentes e essas alterações serão incorporadas em novas edições da publicação. A IBM pode fazer melhorias e/ou alterações no(s) produto(s) e/ou no(s) programa(s) descrito(s) nesta publicação a qualquer momento sem aviso.

Todas as referências nestas informações a websites sites não IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a estes websites sites. Os materiais contidos nesses websites sites não fazem parte dos materiais para este produto IBM e a utilização desses websites sites é de inteira responsabilidade do Cliente.

A IBM pode usar ou distribuir qualquer das informações fornecidas por você da maneira que achar conveniente, sem que isso implique em qualquer obrigação para com o Cliente.

Os licenciados deste programa que desejam obter informações sobre ele para o propósito de ativação: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) o uso mútuo das informações que foram trocadas, devem entrar em contato com:

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

Essas informações podem estar disponível, sujeitas aos termos e condições adequados, incluindo em alguns casos, o pagamento de uma tarifa.

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 de Licença de Programa Internacional IBM ou de qualquer outro contrato equivalente.

Todos os dados de desempenho aqui contidos foram determinados em um ambiente de controle. Assim, 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 estas medidas serão as mesmas em sistemas disponíveis em geral. Além disso, algumas medidas podem ter sido estimadas por meio de extrapolação. Os resultados reais podem variar. Os usuários deste documento devem verificar os dados aplicáveis para seu ambiente específico.

Informações relativas a produtos não IBM foram obtidas dos fornecedores desses produtos, seus anúncios publicados ou outras fontes disponíveis de publicidade. A IBM não testou esses produtos e não pode confirmar a precisão do desempenho, da compatibilidade ou de outras afirmações relacionadas aos produtos não IBM. Perguntas sobre os recursos de produtos não IBM devem ser endereçadas aos fornecedores desses produtos.

Todas as instruções relativas as direções ou intenções futuras da IBM estão sujeitas a mudanças ou retirada sem aviso prévio, e apenas representam metas e objetivos.

Todos os preços IBM mostrados são preços de varejo sugeridos pela IBM, são atuais e estão sujeitos a alterações sem aviso prévio. Os preços do revendedor podem variar.

Estas informações contêm exemplos de dados e de relatórios usados em operações de negócios diárias. Para ilustrá-las como completamente possíveis, os exemplos incluem os nomes dos indivíduos, das empresas, das marcas e dos produtos. Todos

esses nomes são fictícios e qualquer similaridade com nomes e endereços usados por uma empresa real é mera coincidência.

LICENÇA DE COPYRIGHT:

Estas informações contêm programas aplicativos de amostra em idioma de origem, que ilustra técnicas de programação em várias plataformas operacionais. Você pode copiar, modificar e distribuir esses programas de amostra em qualquer formato sem o pagamento à IBM, para os propósitos de desenvolvimento, uso, marketing ou distribuição de programas aplicativos de acordo com a interface de programação de aplicativos para a plataforma operacional para a qual os programas de amostra foram escritos. Esses exemplos não foram completamente testados sob todas as condições. A IBM, dessa forma, não pode garantir ou indicar a confiabilidade, capacidade de manutenção ou função desses programas. Os programas de amostra são fornecidos "no estado em que se encontra", sem garantia de tipo algum. A IBM não será responsável por quaisquer danos decorrentes do uso pelo Cliente dos programas de amostra.

Se estiver vendo estas informações em cópia eletrônica, as fotografias e ilustrações coloridas podem não aparecer.

Marcas Registradas

IBM, o logotipo da IBM e ibm.com são marcas e/ou marcas registradas da International Business Machines Corp., registradas em muitas jurisdições em todo o mundo. Outros produtos e nomes de serviços podem ser marcas registradas da IBM ou de outras empresas. Uma lista atual de marcas registradas da IBM está disponível na Web em "Copyright and trademark information" em www.ibm.com/legal/copytrade.shtml.



Impresso no Brasil