

Versão 9 Release 1.2  
23 de Setembro de 2015

*IBM Marketing Operations Módulo de  
Integração*

**IBM**

**Observação**

Antes de usar estas informações e o produto suportado por elas, leia as informações em “Avisos” na página 31.

Esta edição se aplica à versão 9, liberação 1, modificação 2 do IBM Marketing Operations e a todas as liberações e modificações subsequentes até que seja indicado de outra forma em novas edições.

© Copyright IBM Corporation 2002, 2015.

---

# Índice

## Capítulo 1. O que são Serviços de Integração do IBM Marketing Operations?

Quais são os requisitos para Serviços de Integração do Marketing Operations? . . . . .	2
Informações básicas de Serviços de Integração do IBM Marketing Operations . . . . .	3
Instalando Serviços de Integração . . . . .	5
Conteúdo do software developer kit . . . . .	6
JavaDocs hospedados . . . . .	6
Documentação e ajuda do Marketing Operations ..	7

## Capítulo 2. Serviço da Web de Integração do Marketing Operations ..

WSDL de Serviços de Integração do Marketing Operations . . . . .	9
executeProcedure . . . . .	9
Tipos de dados do Serviço da Web de Integração do Marketing Operations . . . . .	10

## Capítulo 3. Procedimentos do IBM Marketing Operations . . . . .

Suposições. . . . .	13
Parâmetros de configuração . . . . .	15
Design . . . . .	15

Ciclo de vida do procedimento . . . . .	16
Principais classes Java . . . . .	17
Bloqueio de dados . . . . .	18
Transações de procedimento. . . . .	18
Comunicação de procedimento . . . . .	19
Criação de log de procedimento . . . . .	19
Arquivo de definição de plug-in do procedimento	19

## Capítulo 4. API do IBM Marketing Operations . . . . .

Conteúdo da API do IBM Marketing Operations ..	21
Interfaces da API . . . . .	21
Exceções comuns . . . . .	22
Identificadores . . . . .	22
Mapa de atributos . . . . .	24
Tipos de dados enumerados . . . . .	25

## Antes de entrar em contato com o suporte técnico da IBM . . . . .

## Avisos . . . . .

Marcas Comerciais . . . . .	33
Política de Privacidade e Termos de Considerações de Uso . . . . .	33



---

## Capítulo 1. O que são Serviços de Integração do IBM Marketing Operations?

Os Serviços de Integração do IBM®Marketing Operations combinam o Serviço da Web de Integração do Marketing Operations, procedimentos da API e acionadores para estender recursos de negócios.

Os Serviços de Integração do IBM Marketing Operations são um composto do seguinte.

- Serviço da Web de Integração do **Marketing Operations**

Os Serviços de Integração oferecem uma maneira para clientes do Marketing Operations e o IBM Professional Services integrarem o Marketing Operations com outros aplicativos que são executados em seu ambiente.

- Procedimentos e API do **Marketing Operations**

Os procedimentos customizados podem ser definidos no Marketing Operations para estender a lógica de negócios do Marketing Operations de formas arbitrárias. Após definir procedimentos, esses procedimentos podem ser os destinos para as chamadas do serviço da web de Serviços de Integração de outros aplicativos. Os procedimentos também podem ser definidos para enviar mensagens para outros aplicativos.

- **Marketing Operations acionadores**

Os acionadores podem ser associados a eventos e procedimentos no Marketing Operations. Quando esse tipo de evento ocorrer, o acionador associado será executado.

### Versões e compatibilidade com versões anteriores

As futuras versões dos serviços de integração serão compatíveis com versões anteriores com todas as liberações secundárias e de manutenção que compartilham um número da versão principal. No entanto, a IBM reserva o direito de interromper a compatibilidade com uma versão anterior para liberações principais ponto zero (x.0) se o caso de negócios ou técnico garantir.

O número da versão principal dessa API será aumentado se forem feitas algumas das seguintes mudanças.

- Mudanças na interpretação de dados
- Mudanças na lógica de negócios (por exemplo, mudanças em funções de método de serviço)
- Mudanças em parâmetros de método, em tipos de retorno ou em ambos

O número da versão secundária da API será aumentado se forem feitas algumas das seguintes mudanças. Essas mudanças são compatíveis com uma versão anterior por definição.

- Novo método incluído
- Um novo tipo de dados foi incluído e seu uso é restrito a um novo método
- Incluído novo elemento em um tipo enumerado
- Uma nova versão de uma interface é definida com um sufixo de versão

## Autenticação

A autenticação não é necessária; todos os clientes estão associados a um conhecido usuário do IBM Marketing Operations chamado PlanAPIUser. Um administrador do sistema configura os recursos de segurança deste usuário especial para atender às necessidades de todos os clientes de serviço da web.

## Localidade

O único código de idioma suportado é o código de idioma que está configurado atualmente para a instância do sistema do IBM Marketing Operations. Todos os dados dependentes do código de idioma, como mensagens e moeda, são considerados como estando no código de idioma do sistema.

## Gerenciamento de estado

A API e o serviço da web são *stateless*; as informações por cliente não são salvas pela implementação de serviço em chamadas API. Esse recurso fornece uma implementação de serviço eficiente e simplifica o suporte de cluster.

## Transações do banco de dados

Os Serviços de Integração do Marketing Operations não mostram transações do banco de dados para o cliente, mas usam essas informações se estiverem incluídas no contexto do execução. Se uma transação for iniciada, o efeito de todas as chamadas API em um procedimento específico será atômico. Em outras palavras, uma chamada API com falha deixa o banco de dados no mesmo estado, como se a API nunca tivesse sido chamada. Outros usuários do Marketing Operations não verão as mudanças até que o procedimento tenha concluído a transação com sucesso.

As chamadas API que atualizam o banco de dados primeiro devem adquirir um bloqueio de edição para evitar que outros usuários modifiquem os dados subjacentes durante as chamadas API. Outros usuários não podem atualizar componentes bloqueados até que a chamada API seja concluída. Da mesma forma, o próximo usuário ou cliente da API do Marketing Operations deve adquirir o bloqueio dos dados antes do envio de outra chamada API.

## Processamento de eventos

As operações em componentes do IBM Marketing Operations por meio da API geram os mesmos eventos, como se um usuário do Marketing Operations tivesse executado a operação. Os usuários assinantes de algumas notificações, como mudanças de estado do projeto, são notificados de mudanças de estado resultantes de chamadas API e ações do usuário.

---

## Quais são os requisitos para Serviços de Integração do Marketing Operations?

Os Serviços de Integração do Marketing Operations possuem os requisitos a seguir.

Os Serviços de Integração do Marketing Operations devem:

- Acoplar fracamente a integração do sistema.
- Fornecer um mecanismo para aplicativos do cliente para afetar o Marketing Operations por meio de chamadas de serviço da web.

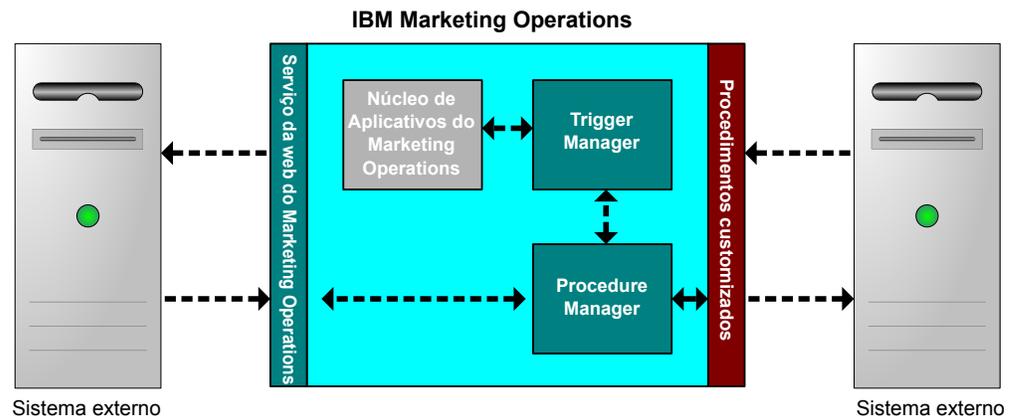
- Fornecer um mecanismo para aplicativos do cliente serem notificados de alguns eventos no Marketing Operations.
- Fornecer um modelo de programação simples que seja fácil de entender e usar.
- Ser robusto durante a recuperação de falha.
- Garantir a integridade de dados.
- Integrar-se com e minimizar o efeito em clientes baseados na GUI do Marketing Operations existentes.
- Fornecer acesso de baixa granularidade a componentes do Marketing Operations enquanto isola programadores de detalhes de implementação subjacentes.

## Informações básicas de Serviços de Integração do IBM Marketing Operations

Use os Serviços de Integração do IBM Marketing Operations para criar procedimentos customizados. É possível usar esses procedimentos para acionar eventos externos quando ocorrerem determinados eventos no Marketing Operations. É possível usar esses procedimentos para executar funções do Marketing Operations a partir de sistemas ou programas externos.

A interface da API interage com o IBM Marketing Operations em nível programático, da mesma forma que a GUI interage com o Marketing Operations em nível do usuário. Usando a API, você constrói os procedimentos. Usando esses procedimentos, você se comunica entre o Marketing Operations e sistemas externos. O Serviço da Web do Marketing Operations é o objeto contêiner para os procedimentos, a API e acionadores.

A arquitetura dos Serviços de Integração do Marketing Operations é mostrada aqui.

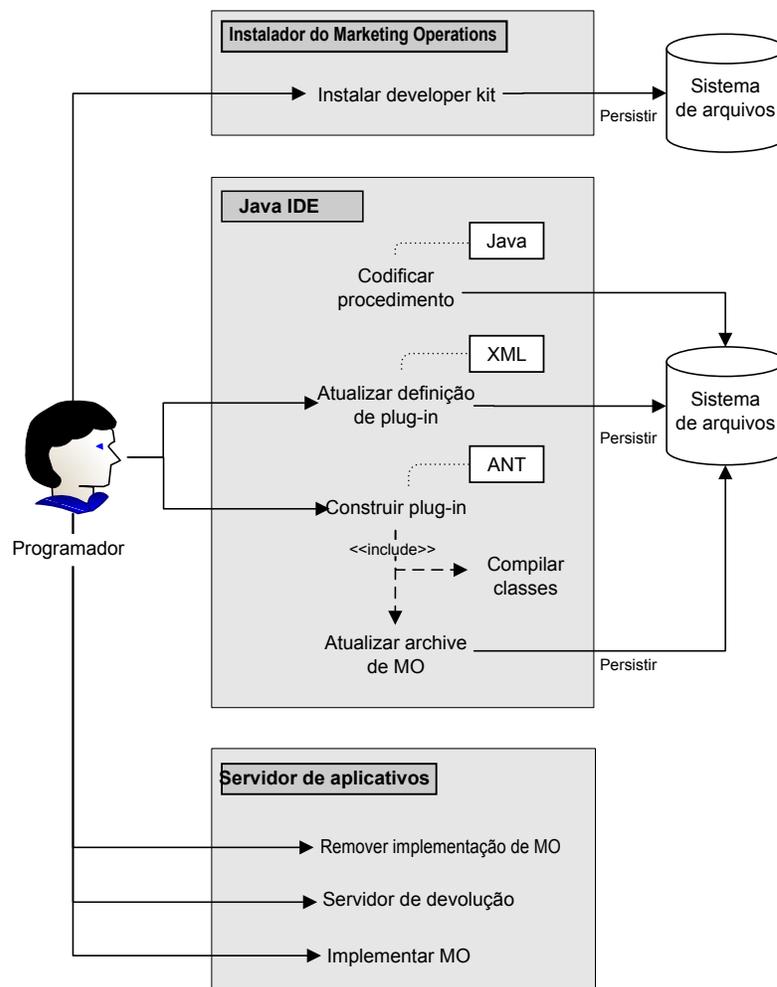


A seguir estão os principais componentes dos Serviços de Integração.

- Marketing Operations Procedure Manager: estende a lógica de negócios interagindo com o Marketing Operations por meio da API.
- Marketing Operations Trigger Manager: associa uma condição (por exemplo, a mudança de estado de um objeto de marketing) com uma ação (um procedimento a ser executado quando a condição para o acionador for atendida).

### Métodos

Use os componentes de Serviços de Integração do IBM Marketing Operations para desenvolver procedimentos customizados, conforme mostrado no diagrama a seguir.



Depois de instalar o kit do desenvolvedor, siga estas etapas básicas:

1. Codifique o procedimento customizado. Atualmente, você deve usar Java™.
2. Atualize a definição do plug-in no arquivo de definição XML.
3. Construa o plug-in:
  - a. Compile as classes necessárias.
  - b. Se estiver usando uma biblioteca de terceiros que não está no archive do Marketing Operations, inclua a biblioteca no arquivo **plan.war** e reimplente.
4. Reinicie o Marketing Operations. As mudanças nas classes de procedimento são aplicadas ao reiniciar o servidor de aplicativos.

**Nota:** Se você alterar o arquivo **plan.war**, deverá remover a implementação e reimplentar o Marketing Operations com o novo arquivo **plan.war**. Remova a implementação e reimplente o Marketing Operations se você usar uma biblioteca de terceiros que não está no archive do Marketing Operations e edite o arquivo **plan.war**.

## Exemplo básico para comunicação entre o IBM Marketing Operations e a API

O exemplo básico a seguir descreve como estabelecer comunicação entre a API e o Marketing Operations. Ele não executa nenhum trabalho útil; executa um roundtrip entre o Marketing Operations e os Serviços de Integração.

Esse exemplo usa partes dos procedimentos de exemplo incluídos com o kit do desenvolvedor dos Serviços de Integração do Marketing Operations. Especificamente, é possível localizar o código referenciado aqui nos seguintes arquivos.

- PlanClientFacade.java
- PlanWSNOOPTestCase.java

O método noop é uma chamada de serviço da web para o Marketing Operations. Ele é definido na classe PlanClientFacade e transmite os valores nulos em uma matriz.

```
public ProcedureResponse noop(String jobId)
    throws RemoteException, ServiceException {
    NameValueArrays parameters =
        new NameValueArrays(null, null, null, null, null, null, null, null);
    return _serviceBinding.executeProcedure("uapNOOPProcedure", jobId, parameters);
}
```

O procedimento testExecuteProcedure chama o método noop a partir de PlanClientFacade para estabelecer um roundtrip com o aplicativo Marketing Operations.

```
public void testExecuteProcedure() throws Exception {
    // Atingir tempo limite após um minuto
    int timeout = 60000;
    PlanClientFacade clientFacade = new PlanClientFacade(urlWebService, timeout);
    System.out.println("noop w/no parameters");
    long startTime = new Date().getTime();
    ProcedureResponse response = clientFacade.noop("junit-jobid");
    long duration = new Date().getTime() - startTime;

    // status zero ou positivo => sucesso
    System.out.println("Status: " + response.getStatus());
    System.out.println("Duration: " + duration + " ms");
    assertTrue(response.getStatus() >= 0);
    System.out.println("Done.");
}
```

Para obter detalhes de NameValueArrays, ProcedureResponse e outros métodos e tipos de dados listados, consulte o *Marketing Operations Integration Module* e os JavaDocs.

## Instalando Serviços de Integração

O módulo Serviços de Integração do IBM Marketing Operations é um componente separado e pago. Se você comprar o módulo Serviços de Integração, deverá instalá-lo.

1. Faça download dos instaladores dos Serviços de Integração do IBM Marketing Operations.
2. Os instaladores do IBM EMM detectam o módulo Serviços de Integração.
3. O instalador configura as propriedades de configuração em **Marketing Operations | umConfiguration | integrationServices |**

**enableIntegrationServices.** É possível customizar sua instalação alterando parâmetros de configuração. Para obter mais informações, consulte “Parâmetros de configuração” na página 15.

## Conteúdo do software developer kit

Geralmente, todos os componentes de Serviços de Integração do Marketing Operations estão instalados em uma pasta chamada "devkits."

O software developer kit contém documentação, Javadoc contendo todas as classes e interfaces publicapi e código de exemplo.

O código de exemplo é instalado nas seguintes pastas.

- A pasta **build** contém scripts para construir e implementar procedimentos customizados.

- A pasta **Classes** contém as classes de procedimento compiladas.

Os usuários devem implementar as classes compiladas de seus procedimentos customizados no caminho especificado pelo parâmetro de configuração

**integrationProcedureClasspathURL.** Em seguida, o IBM Marketing Operations Procedure Manager carrega-as conforme especificado no arquivo de configuração `procedure-plugins.xml`.

- A pasta **lib** contém as bibliotecas necessárias para desenvolver e compilar procedimentos customizados.

- A pasta **src** contém arquivos de origem para procedimentos customizados. Os usuários podem colocar procedimentos customizados a serem iniciados como acionadores ou serviços da web aqui.

- A pasta **src/procedure** contém o arquivo de configuração `procedure-plugins.xml`. Cada procedimento customizado que é executado como um acionador baseado em um evento ou por meio de um serviço da web externo deve ter uma entrada nesse arquivo. As entradas devem conter um caminho de classe completo de procedimento e parâmetros de inicialização obrigatórios.

- A pasta **src/procedure** também contém alguns procedimentos de amostra que estão incluídos no IBM Marketing Operations. Esses procedimentos podem ser usados para compreender e desenvolver seus procedimentos customizados.

Coloque os procedimentos customizados no diretório **src** em uma nova estrutura de pasta, como `com/<mycompany>/<mypackage>`. Não coloque os procedimentos customizados na pasta `src/procedures` de amostra.

- A pasta **src/soap** contém os clientes de serviço da web de amostra que são desenvolvidos em Java. Use essas amostras como um ponto de início para desenvolver clientes baseados em serviço da web para Serviços de Integração. Esta pasta também contém scripts binários para iniciar os clientes de amostra através da linha de comandos.

---

## JavaDocs hospedados

Para obter informações específicas sobre os métodos da API públicos, consulte a classe `iPlanAPI` nos arquivos de documentação da API de JavaDocs.

Esses arquivos estão disponíveis nas seguintes maneiras:

- Pelos arquivos no diretório `<IBM_EMM>/<MarketingOperations_Home>/devkits/integration/javadocs` no servidor que hospeda o Marketing Operations.

- Efetuando login no Marketing Operations e selecionando **Ajuda > Documentação do produto** a partir de qualquer página e, em seguida, fazendo download do arquivo IBM <version>PublicAPI.zip.

## Documentação e ajuda do Marketing Operations

Pessoas diferentes de sua organização usam o IBM Marketing Operations para cumprir tarefas diferentes. Informações sobre o Marketing Operations estão disponíveis em um conjunto de guias, cada um dos quais se destina ao uso por membros da equipe com objetivos e conjuntos de qualificações específicos.

A tabela a seguir descreve as informações disponíveis em cada guia.

*Tabela 1. Guias do Conjunto de Documentação do Marketing Operations.*

A tabela de três colunas a seguir descreve tarefas em uma coluna, os nomes de guias na segunda coluna e o público na terceira coluna.

Se você	Consulte	Público
<ul style="list-style-type: none"> <li>• Planeje e gerencie projetos</li> <li>• Estabeleça tarefas de fluxo de trabalho, marcos e equipe</li> <li>• Rastrear despesas do projeto</li> <li>• Obtenha revisões e aprovações para o conteúdo</li> <li>• Produza relatórios</li> </ul>	<i>IBM Marketing Operations Guia do Usuário</i>	<ul style="list-style-type: none"> <li>• Coordenadores de projeto</li> <li>• Designers criativos</li> <li>• Gerentes de marketing de correio direto</li> </ul>
<ul style="list-style-type: none"> <li>• Modelos de design, formulários, atributos e métricas</li> <li>• Customize a interface com o usuário</li> <li>• Defina os níveis de acesso do usuário e a segurança</li> <li>• Implemente recursos opcionais</li> <li>• Configure e ajuste o Marketing Operations</li> </ul>	<i>IBM Marketing Operations Guia do Administrador</i>	<ul style="list-style-type: none"> <li>• Coordenadores de projeto</li> <li>• Administradores de TI</li> <li>• Consultores de implementação</li> </ul>
<ul style="list-style-type: none"> <li>• Criar campanhas de marketing</li> <li>• Planejar ofertas</li> <li>• Implementar integração entre Marketing Operations e Campaign</li> <li>• Implementar integração entre Marketing Operations e IBM Digital Recommendations</li> </ul>	<i>IBM Marketing Operations and IBM Integration Guide</i>	<ul style="list-style-type: none"> <li>• Coordenadores de projeto</li> <li>• Especialistas de execução de marketing</li> <li>• Gerentes de marketing direto</li> </ul>
<ul style="list-style-type: none"> <li>• Aprenda sobre novos recursos do sistema</li> <li>• Pesquise problemas conhecidos e soluções alternativas</li> </ul>	<i>Notas sobre a Liberação do IBM Marketing Operations</i>	Todos os que usam o Marketing Operations
<ul style="list-style-type: none"> <li>• Instale o Marketing Operations</li> <li>• Configure o Marketing Operations</li> <li>• Faça upgrade para uma nova versão do Marketing Operations</li> </ul>	<i>IBM Marketing Operations Guia de Instalação</i>	<ul style="list-style-type: none"> <li>• Consultores de implementação de software</li> <li>• Administradores de TI</li> <li>• Administradores de banco de dados</li> </ul>

Tabela 1. Guias do Conjunto de Documentação do Marketing Operations (continuação).

A tabela de três colunas a seguir descreve tarefas em uma coluna, os nomes de guias na segunda coluna e o público na terceira coluna.

Se você	Consulte	Público
Crie procedimentos customizados que integrem o Marketing Operations com outros aplicativos	<i>IBM Marketing Operations Integration Module</i> e os JavaDocs da API disponíveis ao clicar em <b>Ajuda &gt; Documentação do Produto</b> em Marketing Operations e, em seguida, fazer download do arquivo <code>IBM&lt;version&gt;PublicAPI.zip</code>	<ul style="list-style-type: none"> <li>• Administradores de TI</li> <li>• Administradores de banco de dados</li> <li>• Consultores de implementação</li> </ul>
Aprenda sobre a estrutura do banco de dados Marketing Operations	<i>IBM Marketing Operations System Schema</i>	Administradores de banco de dados
Precisa de mais informações enquanto trabalha?	<ul style="list-style-type: none"> <li>• Obtenha ajuda e procure ou navegue pelos guias do <i>Usuário e do Administrador do Marketing Operations</i> ou de <i>Instalação do Marketing Operations</i>: Clique em <b>Ajuda &gt; Ajuda para essa página</b></li> <li>• Acesse todos os guias Marketing Operations: Clique em <b>Ajuda &gt; Documentação do Produto</b></li> <li>• Guias de acesso para todos os produtos do IBM Enterprise Marketing Management (EMM): Clique em <b>Ajuda &gt; Toda a Documentação do Conjunto IBM EMM</b></li> </ul>	Todos os que usam o Marketing Operations

---

## Capítulo 2. Serviço da Web de Integração do Marketing Operations

O serviço da web fornece uma visualização de cliente dos Serviços de Integração do Marketing Operations, que faz parte da implementação do servidor do IBM Marketing Operations. O serviço é usado simultaneamente com usuários da web do Marketing Operations.

O serviço da web suporta uma chamada API, `executeProcedure`.

Um cliente faz esta chamada de serviço da web diretamente.

---

### WSDL de Serviços de Integração do Marketing Operations

O Web Services Definition Language (WSDL) foi definido manualmente e é a palavra final na definição de serviço da web.

#### Axis

Essa versão do serviço da web usa o Axis2 1.5.2 para gerar classes do lado do servidor que formam a implementação do serviço da web a partir do arquivo WSDL. Os usuários podem usar qualquer versão do Axis, ou uma técnica não Axis para criar uma implementação do lado do cliente para integração com a API a partir do WSDL fornecido.

#### Versão de protocolo

A versão do protocolo está explicitamente ligada ao WSDL, da seguinte forma:

- Como parte do nome WSDL, por exemplo, `PlanIntegrationService1.0.wsdl`
- Como parte do `targetNamespace` WSDL, por exemplo, `xmlns:tns="http://webservices.unica.com/MktOps/services/PlanIntegrationServices1.0?wsdl"`

#### WSDL

Um arquivo WSDL é fornecido com os Serviços de Integração do IBM Marketing Operations: `PlanIntegrationServices1.0.wsdl`. O WSDL é entregue no diretório `integration/examples/soap/plan`. O script de construção de exemplo usa esse arquivo para gerar os stubs do lado do cliente apropriados para conectar-se ao serviço da web.

---

### `executeProcedure`

`executeProcedure` é a chamada API que é suportada pelo serviço da web.

#### Sintaxe

```
executeProcedure(string key, string jobid, NameValueArrays paramArray)
```

#### Retorna

```
int: status  
Message[]: messages
```

## Descrição

Esse método chama o procedimento especificado com uma matriz opcional de parâmetros. A chamada é executada de forma síncrona; ou seja, ela bloqueia o cliente e retorna o resultado após a conclusão.

## Parâmetros

Tabela 2. Parâmetros executeProcedure

Nome	Descrição
chave	A chave exclusiva do procedimento a ser executado. Será retornado um erro <i>RemoteException</i> se nenhum procedimento estiver ligado à <b>chave</b> .
jobid	Sequência opcional que identifica a tarefa que está associada a essa execução de procedimento. Essa sequência é um item de passagem, mas pode ser usada para ligar tarefas do cliente à execução de um determinado procedimento.
paramArray	Uma matriz de parâmetros a serem transmitidos para o procedimento. Um status e uma mensagem de erro serão retornados se um ou mais dos parâmetros for inválido (por exemplo, tipo errado ou um valor incorreto). É o cliente que determina os parâmetros, seus tipos e o número de valores que são requeridos pelo procedimento.

## Parâmetros de retorno

Tabela 3. Parâmetros de retorno executeProcedure

Nome	Descrição
status	Um código de número inteiro: <ul style="list-style-type: none"><li>• 0 indica que o procedimento foi executado com sucesso</li><li>• um número inteiro indica um erro</li></ul> Os procedimentos podem usar o status para indicar diferentes níveis de erros.
mensagens	Uma matriz de zero ou mais estruturas de dados da mensagem. Se <b>status</b> for 0, essa matriz não conterá mensagens ERROR, mas poderá conter mensagens INFORMATION e WARNING.  Se <b>status</b> for diferente de zero, as mensagens podem conter qualquer combinação de mensagens ERROR, INFORMATION e WARNING.

---

## Tipos de dados do Serviço da Web de Integração do Marketing Operations

Os tipos de dados que são usados pelo serviço da web são independentes de qualquer implementação de ligação ou de programação de serviço específica.

A seguinte notação é usada.

- `<type>`: `<type definition>` define um tipo de dados simples. Por exemplo:  
Identificador: sequência
- `<type>`: `[ <type definition> ]` define um tipo de dados complexo ou uma estrutura de dados.
- `<type>`: `{ <type definition> }` define um tipo de dados complexo ou uma estrutura de dados.

Os elementos de tipo complexo e os parâmetros da API podem usar esses tipos para declarar matrizes. Por exemplo:

Handle [] handles

O tipo, identificadores, é uma matriz de tipos de Identificador.

## Tipos primitivos

Os tipos primitivos são restritos aos tipos definidos na tabela a seguir para simplificar o suporte para ligações SOAP 1.1. Todos os tipos podem ser declarados

como matrizes, por exemplo, **String** [ ]. De forma inerente, os tipos de dados binários, como **long**, podem ser representados como sequências por uma ligação de protocolo (por exemplo, SOAP). No entanto, essa representação não tem efeito na semântica do tipo, nos valores permitidos, etc., conforme vistos pelo cliente.

*Tabela 4. Tipos primitivos*

Tipo de API	Descrição	Tipo SOAP	Tipo Java
Boolean	Valor booleano: <b>true</b> ou <b>false</b>	xsd:Boolean	Boolean
dateTime	Um valor de data e hora	xsd:dateTime	Data
decimal	Um valor decimal de precisão arbitrária	xsd:decimal	java.math.BigDecimal
double	Um valor decimal, assinado, de precisão dupla	xsd:double	double
int	Um valor de número inteiro assinado, de 32 bits	xsd:int	int
integer	Um valor de número inteiro assinado, de precisão arbitrária	xsd:integer	java.math.BigInteger
long	Um valor de número inteiro assinado, de 64 bits	xsd:long	long
string	Uma sequência de caracteres Unicode	xsd:string	java.lang.String

## MessageTypeEnum

MessageTypeEnum: { INFORMATION, WARNING, ERROR }

MessageTypeEnum é um tipo enumerado que define todos os tipos de mensagens possíveis.

- INFORMATION: uma mensagem informativa
- WARNING: uma mensagem de aviso
- ERROR: uma mensagem de erro

## Mensagem

Message: [MessageTypeEnum type, string code, string localizedText, string logDetail]

Mensagem é uma estrutura de dados que define o resultado de uma chamada API de serviço da web. Ela fornece campos opcionais para um código não localizado, texto localizado e detalhes do log. Atualmente, todo o texto localizado usa o código de idioma que está configurado para a instância de servidor do IBM Marketing Operations.

*Tabela 5. Parâmetros da mensagem*

Parâmetro	Descrição
tipo	Um MessageTypeEnum, configurar o tipo da mensagem.
code	Um código opcional, no formato de sequência, para a mensagem.
localizedText	Uma sequência de texto opcional a ser associada à mensagem.
logDetail	Um mensagens de rastreamento de pilha opcional.

## NameValue

NameValue: [string name, int sequence]

NameValue é um tipo complexo base que define um par nome-valor. Ele também define uma sequência opcional que o serviço usa para construir matriz de valor conforme necessário (as sequências são baseadas em zero).

Todos os NameValues com o mesmo nome, mas números de sequência diferentes, são convertidos em uma matriz de valores e associados ao nome comum.

O tamanho da matriz é determinado pelo número de sequência máximo; os elementos da matriz não especificados possuem valores nulos. Os números de sequência da matriz devem ser exclusivos. O valor e seu tipo são fornecidos pelo tipo estendido.

*Tabela 6. Parâmetros NameValue*

Parâmetro	Descrição
nome	Uma sequência que define o nome de um tipo NameValue.
sequence	Um número inteiro baseado em zero que configura o número de sequência para o valor implícito NameValue.

Os tipos estendidos NameValue são definidos para cada tipo primitivo, da seguinte forma:

*Tabela 7. Tipos estendidos NameValue*

Tipo estendido	Descrição
BigDecimalNameValue: NameValue [ decimal value]	Um tipo NameValue cujo valor é um número decimal de precisão arbitrária.
BigIntegerNameValue: NameValue [ integer value]	Um tipo NameValue cujo valor é um número inteiro dimensionado de forma arbitrária.
BooleanNameValue: NameValue [ Boolean value]	Um tipo NameValue cujo valor é um Booleano.
CurrencyNameValue: NameValue [ string locale, decimal value]	Um tipo NameValue adequado para representar a vigência em algum código de idioma. Código de idioma é um código de idioma ISO, ou seja, os códigos de duas letras minúsculas, conforme definido por ISO-639.  Atualmente, o código de idioma deve estar de acordo com o código de idioma configurado na instância de servidor do IBM Marketing Operations.
DateNameValue: NameValue [ datetime value]	Um tipo NameValue cujo valor é uma data.
DecimalNameValue: NameValue [ double value]	Um tipo NameValue cujo valor é um número decimal de precisão dupla.
IntegerNameValue: NameValue [ long value]	Um tipo NameValue cujo valor é um número inteiro de 64 bits.
String NameValue: NameValue [ string value]	Um tipo NameValue cujo valor é uma sequência.

E por último, uma matriz dos tipos estendidos NameValue é definida para uso quando for necessário definir um conjunto de NameValues com tipos diferentes.

```

NameValueArrays: [
  BooleanNameValue[]    booleanValues,
  StringNameValue[]     stringValue,
  IntegerNameValue[]    integerValue,
  BigIntegerNameValue[] bigIntegoooleanNameValue,
  DecimalNameValue[]    decimalValues,
  BigDecimalNameValue[] bigDecimalValues
  DateNameValue[]      dateNameValues
  CurrencyNameValue[]   currencyValues
]

```

---

## Capítulo 3. Procedimentos do IBM Marketing Operations

Um "procedimento" é uma classe Java customizada ou padrão hospedada pelo IBM Marketing Operations que executa alguma unidade de trabalho. Os procedimentos fornecem uma maneira para os clientes e o IBM Professional Services estenderem a lógica de negócios do Marketing Operations de formas arbitrárias.

Os procedimentos seguem um modelo de programação simples com uma API bem definida para afetar os componentes que são gerenciados pelo Marketing Operations. Os procedimentos são "descobertos" por meio de um mecanismo de consulta simples e arquivo de definição baseado em XML. O Marketing Operations executa os procedimentos de acordo com as necessidades de seus "clientes". Por exemplo, em resposta a uma solicitação de integração (recebida) ou a um disparo de acionador (interno ou de saída).

Os procedimentos são executados de forma síncrona com seus clientes; os resultados são disponibilizados diretamente para o cliente e por meio de um mecanismo de auditoria persistido. A execução de um procedimento também pode fazer outros eventos e acionadores serem disparados no Marketing Operations.

Os procedimentos devem ser gravados em Java.

---

### Suposições

As classes de implementação do procedimento são empacotadas em uma árvore de classes ou arquivo JAR separado e disponibilizadas para o IBM Marketing Operations por meio de um caminho de URL.

#### Implementação de procedimento

O gerenciador de execução do procedimento usa um carregador de classes independente para carregar estas classes, conforme necessário. Por padrão, o Marketing Operations examina o diretório a seguir.

```
<MarketingOperations_Home>/devkits/integration/examples/classes
```

Para alterar esse padrão, configure o parâmetro **integrationProcedureClasspathURL** em **Configurações > Configuração > Marketing Operations > umoConfiguration > integrationServices**.

O nome da classe de implementação do procedimento segue as convenções de nomenclatura Java aceitas para evitar colisões de pacotes com "unica" e classes de outros fornecedores. Em específico, os clientes não devem colocar procedimentos na árvore do pacote "com.unica" ou "com.unicacorp".

A implementação do procedimento é codificada para a versão de Java runtime usada pelo IBM Marketing Operations no servidor de aplicativos (pelo menos o JRE 1.5.10).

A classe de implementação do procedimento é carregada pela política de carregamento de classe que normalmente é usada pelo IBM Marketing Operations (geralmente **pai por último**). O servidor de aplicativos pode fornecer ferramentas de desenvolvimento e as opções para recarregar as classes que se aplicariam aos

procedimentos do Marketing Operations, mas isso não é necessário.

## Bibliotecas

O IBM Marketing Operations fornece algumas bibliotecas de software livre e de terceiros; os servidores de aplicativos também usam versões diferentes dessas bibliotecas.

Em geral, esta lista muda de liberação para liberação. Para a versão 9.1.0, as seguintes bibliotecas de terceiros são suportadas.

- Ant 1.6.5 (ant.jar)
- Axis2 1.5.2 e dependências
  - axiom-api-1.2.9.jar
  - axiom-impl-1.2.9.jar
  - axis2-adb-codegen-1.5.2.jar
  - axis2-codegen-1.5.2.jar
  - axis2-adb-1.5.2.jar
  - axis2-kernel-1.5.2.jar
  - axis2-transport-http-1.5.2.jar
  - axis2-transport-local-1.5.2.jar
  - commons-codec.jar
  - commons-httpclient-3.1.jar
  - commons-logging.jar
  - httpcore-4.0.jar
  - neethi-2.0.4.jar
  - geronimo-stax-api\_1.0\_spec-1.0.1.jar
  - jaxrpc.jar
  - xlxpScanner.jar
  - xlxpScannerUtils.jar
  - xlxpWASParsers.jar
  - wsdl4j-1.6.2.jar
  - XmlSchema-1.4.3.jar
- JavaMail 1.4.3 (activation.jar, mail.jar)
- JUnit 4.4 (junit-4.4.jar)
- APIs do IBM Marketing Operations (affinium\_plan.jar)
- APIs do IBM Marketing Platform (unica-common.jar)

Se um procedimento, ou as classes secundárias importadas pelo procedimento usarem esses pacotes, seu uso deve concordar exatamente com os pacotes fornecidos pelo Marketing Operations ou pelo servidor de aplicativos. Nesse caso, o retrabalho de seu código de procedimento será necessário se uma versão mais recente do Marketing Operations fizer upgrade ou abandonar uma biblioteca.

## Procedimentos e encadeamentos

O procedimento deve ser thread-safe com relação a seu próprio estado; ou seja, seu método de execução não pode depender de mudanças de estado internas de chamada para chamada. Um procedimento não pode criar encadeamentos por conta própria.

---

## Parâmetros de configuração

Ao instalar o Módulo de Integração do Marketing Operations, o instalador configura três propriedades de configuração. É possível modificar as propriedades de configuração para customizar o comportamento do Módulo de Integração.

As propriedades de configuração para o Módulo de Integração estão em **Marketing Operations | umoConfiguration | integrationServices**.

- A propriedade de configuração **enableIntegrationServices** para ativar e desativar o módulo Serviços de Integração.
- O parâmetro **integrationProcedureDefinitionPath** contém o caminho de arquivo completo para o arquivo XML de definição do procedimento customizado.  
O valor padrão é `<IBM_EMM_Home><MarketingOperations_Home>/devkits/integration/examples/src/procedure/procedure-plugins.xml/`.
- O parâmetro **integrationProcedureClasspathURL** contém a URL para o caminho de classe para procedimentos customizados.  
O valor padrão é `file:///<IBM_EMM_Home><MarketingOperations_Home>/devkits/integration/examples/classes/`.

**Nota:** O '/' no final do caminho `integrationProcedureClasspathURL` é necessário para carregar as classes de procedimento corretamente.

---

## Design

A classe de implementação do procedimento usa a API do IBM Marketing Operations para ler e atualizar componentes, serviços de início do Marketing Operations, etc. Outros pacotes Java podem ser usados para executar outras tarefas.

Em seu design, focalize a produção de uma única unidade de trabalho que opera atômicamente. Teoricamente, um processo executa alguma série de tarefas que podem ser planejadas de forma assíncrona para serem executadas algum tempo depois. Esse modelo de integração "fire and forget" resulta no menor carregamento em ambos os sistemas.

**Nota:** Apenas as classes e métodos documentados serão suportados em futuras liberações do Marketing Operations. Considere todas as outras classes e métodos no Marketing Operations como fora dos limites.

Após codificar e compilar as classes de implementação do procedimento, disponibilize-os para o Marketing Operations. Os scripts de construção que são fornecidos com o Marketing Operations Integration Services colocam os procedimentos compilados no local padrão. A etapa de desenvolvimento final é atualizar o arquivo de definição de plug-in do procedimento customizado que é usado pelo Marketing Operations para descobrir os procedimentos customizados.

O procedimento deve implementar a interface **com.unica.publicapi.plan.plugin.procedure.IProcedure** e ter um construtor sem parâmetro (modelo JavaBeans usual). A codificação e compilação de cada procedimento são feitas em um Java IDE de escolha do cliente, como Eclipse, Borland JBuilder ou Idea. O código de amostra é fornecido com o IBM Marketing Operations como kits de ferramentas do desenvolvedor, no seguinte local:

`<MarketingOperations_Home>/devkits/integration/examples/src/procedure`

---

## Ciclo de vida do procedimento

Cada procedimento é executado por meio de um ciclo de vida completo.

O ciclo de vida de tempo de execução de um procedimento inclui as seguintes etapas.

1. Descoberta e inicialização
2. Seleção (opcional)
3. Execução
4. Destruição

### Descoberta e inicialização

O IBM Marketing Operations deve estar cientes de todos os procedimentos padrão e customizadas disponíveis para uma instância de instalação específica. Esse processo é chamado descoberta.

**Nota:** Procedimentos padrão (procedimentos que são definidos pela equipe de engenharia do Marketing Operations) são conhecidos implicitamente e, portanto, não precisam de nenhuma ação para serem descobertos.

Procedimentos customizados são definidos no arquivo de definição de plug-in do procedimento. O gerenciador de plug-ins do Marketing Operations lê esse arquivo durante a inicialização. Para cada procedimento localizado, o gerenciador de plug-in conclui as seguintes etapas.

1. Instanciar o procedimento; fazer a transição de seu estado para INSTANTIATED.
2. Criar um registro de auditoria do procedimento.
3. Se o procedimento foi instanciado, seu método **initialize()** será chamado com qualquer parâmetro de inicialização localizado em seu arquivo de descrição de plug-in. Se este método lançar uma exceção, o status será registrado e o procedimento será abandonado. Caso contrário, o estado do procedimento será alterado para o estado INITIALIZED. Agora ele está pronto para ser executado.
4. Criar um registro de auditoria do procedimento.
5. Se o procedimento foi inicializado, seu método **getKey()** será chamado para determinar a chave que será usada por clientes para referenciar o procedimento. Essa chave é associada à instância e salva para consulta posterior.

### Seleção

De vez em quando, o IBM Marketing Operations pode apresentar uma lista dos procedimentos disponíveis para os usuários, por exemplo, para permitir que os administradores configurem um acionador. O Marketing Operations apresenta essa lista apenas após o procedimento ter sido inicializado, usando os métodos **getDisplayName()** e **getDescription()** do procedimento.

### Execução

Em algum ponto após a inicialização do procedimento, o IBM Marketing Operations recebe uma solicitação para executar o procedimento. Essa solicitação pode ocorrer simultaneamente com outros procedimentos (ou o mesmo procedimento) em execução em outros encadeamentos.

No tempo de execução, o gerenciador de execução do procedimento conclui as seguintes etapas.

1. Iniciar uma transação do banco de dados.
2. Configurar o estado do procedimento como EXECUTING.
3. Criar um registro de auditoria do procedimento.
4. Chamar o método **execute()** do procedimento com um contexto de execução e quaisquer parâmetros de execução fornecidos pelo cliente. A implementação do método usa a API do Marketing Operations conforme necessário, adquirindo bloqueios de edição e transmitindo o contexto de execução. Se o método de execução lançar uma exceção, o gerenciador de execução marcará a transação para retrocesso.
5. Confirmar ou retroceder a transação, de acordo com os resultados da execução; configurar o estado do procedimento como EXECUTED.
6. Liberar todos os bloqueios de edição pendentes.
7. Criar um registro de auditoria do procedimento.

**Nota:** O método **execute()** não se destina a alterar os dados da instância do procedimento.

## Destruição

Quando o IBM Marketing Operations for encerrado, o gerenciador de plug-in do procedimento percorrerá todos os procedimentos carregados. Para cada procedimento localizado, ele conclui as seguintes etapas.

1. Chama o método `destroy()` do procedimento para permitir que o procedimento seja limpo antes da destruição da instância.
2. Altera o estado do procedimento para FINALIZED (ele não pode ser executado).
3. Cria um registro de auditoria do procedimento.
4. Destrói a instância do procedimento.

---

## Principais classes Java

O developer kit de integração fornecido contém um conjunto de Javadoc para a API pública e classes de suporte do IBM Marketing Operations.

As classes Java mais importantes são listadas aqui.

- `IProcedure` (`com.unica.publicapi.plan.plugin.procedure.IProcedure`): interface que todos os procedimentos devem implementar. Os procedimentos passam por um ciclo de vida bem definido e acessam a API do Marketing Operations para executar o trabalho.
- `ITriggerProcedure` (`com.unica.publicapi.plan.plugin.procedure.ITriggerProcedure`): interface que todos os procedimentos do acionador devem implementar (interface do marcador).
- `IExecutionContext` (`com.unica.publicapi.plan.plugin.procedure.IExecutionContext`): interface do objeto de contexto opaco que é distribuída para o procedimento pelo gerenciador de execução. Esse objeto possui métodos públicos para criação de log e gerenciamento de bloqueio de edição. O procedimento também transmite esse objeto para todas as chamadas PlanAPI.

- IPlanAPI (com.unica.publicapi.plan.api.IPlanAPI): interface para a API do Marketing Operations. O contexto de execução fornece um método **getPlanAPI()** para recuperar a implementação apropriada.

---

## Bloqueio de dados

O IBM Marketing Operations usa um esquema de bloqueio de edição pessimista; ou seja, somente um usuário recebe acesso de atualização a instâncias do componente de cada vez. Para o usuário da GUI, este bloqueio é feito no nível da guia visual. Em alguns casos, os dados são bloqueados para um subconjunto de uma instância, por exemplo, uma guia de resumo do projeto. Em outros casos, os dados são bloqueados em muitas instâncias, por exemplo, a guia Fluxo de trabalho. Após um usuário adquirir um bloqueio, todos os outros usuários ficam restritos ao acesso somente leitura aos dados relacionados.

Para assegurar que as mudanças feitas por um procedimento em uma instância do componente ou grupo de instâncias não sejam sobrescritas acidentalmente por outro usuário, um procedimento deve adquirir os bloqueios apropriados antes de atualizar dados do componente. O objeto de contexto de execução transmitido para o método **execute()** do procedimento é usado para realizar o bloqueio dos dados.

Antes de o procedimento atualizar quaisquer dados, ele deve chamar o método **acquireLock()** do contexto para cada bloqueio necessário. Por exemplo, se um procedimento for atualizar um projeto e o fluxo de trabalho associado, o procedimento deverá adquirir bloqueios para ambos.

Se outro usuário já tiver um bloqueio, o método **acquireLock()** lançará um **LockInUseException** imediatamente. Para minimizar colisões, o procedimento deverá liberar o bloqueio assim que ele atualizar o objeto.

O gerenciador de execução libera automaticamente todos os bloqueios pendentes quando o método de execução retorna. Em qualquer caso, os bloqueios são mantidos apenas pela duração da transação do banco de dados. Ou seja, os bloqueios expirarão se o tempo limite de transação específico do banco de dados for excedido.

**Nota:** Os bloqueios de edição não são iguais a transações do banco de dados.

---

## Transações de procedimento

O gerenciador de execução de procedimento agrupa automaticamente a execução do procedimento com uma transação do banco de dados, confirmando ou retrocedendo-a com base no resultado da execução do procedimento.

O agrupamento de execução do procedimento e de transação do banco de dados assegura que as atualizações no banco de dados do IBM Marketing Operations não fiquem visíveis para outros usuários, até que sejam confirmadas. Ele também torna as atualizações atômicas.

O gravador de procedimento ainda deverá adquirir os bloqueios de edição necessários para assegurar que outros usuários não possam gravar alterações no banco de dados antes da execução do procedimento ser concluída.

---

## Comunicação de procedimento

O método **execute()** de um procedimento retorna um código de status de número inteiro à tabela de auditoria de procedimento do IBM Marketing Operations. O método **execute()** de um procedimento também pode retornar zero ou mais mensagens na tabela de auditoria do procedimento, que são registradas e persistidas.

O cliente também pode comunicar as informações de status de alguma outra maneira.

---

## Criação de log de procedimento

O IBM Marketing Operations possui um arquivo de log separado para procedimentos: `<MarketingOperations_Home>\logs\procedure.log`

O gerenciador de execuções de procedimentos registra o ciclo de vida de cada procedimento e cria registros de auditoria.

- **logInfo()**: gravar uma mensagem informativa no log de procedimento.
- **logWarning()**: gravar uma mensagem de aviso no log de procedimento.
- **logError()**: gravar uma mensagem de erro no log de procedimento.
- **logException()**: fazer dump do rastreamento de pilha para a exceção no log de procedimento.

---

## Arquivo de definição de plug-in do procedimento

O arquivo de definição de plug-in de procedimento define a classe de implementação, os metadados, e outras informações sobre os procedimentos customizados a serem hospedados no IBM Marketing Operations.

Por padrão, a definição do plug-in de procedimento é considerada como estando no seguinte caminho:

```
<MarketingOperations_Home>/devkits/integration/examples/src/procedures/  
procedure-plugins.xml
```

Este arquivo é um documento XML que contém as informações a seguir.

Procedimentos: uma lista de zero ou mais elementos **Procedure**.

Procedimento: um elemento que define um procedimento. Cada procedimento contém os elementos a seguir.

- **key** (opcional): sequência que define a chave de consulta para o procedimento. Essa chave deve ser exclusiva entre todos os procedimentos padrão (fornecidos pelo IBM) e customizados que são hospedados por uma determinada instância do Marketing Operations. Se não definido, será padronizado para a versão completa do elemento **className**. Nomes que começam com a sequência "uap" são reservados para serem usados pelo IBM Marketing Operations.
- **className** (obrigatório): nome completo do pacote da classe de procedimento. Esta classe deve implementar a classe `IProcedure` (`com.unica.public.plan.plugin.procedure.IProcedure`).
- **initParameters** (opcional): uma lista de zero ou mais elementos `initParameter`.

**initParameter** (opcional): parâmetro a ser transmitido para o método initialize() do procedimento. Esse elemento inclui o nome, tipo e elementos de valor do parâmetro aninhado.

- name: sequência que define o nome do parâmetro
- type: nome de classe opcional da classe do wrapper Java que define o tipo do valor de parâmetro. Deve ser um dos seguintes tipos:
  - java.lang.String (o padrão)
  - java.lang.Integer
  - java.lang.Double
  - java.lang.Calendar
  - java.lang.Boolean
- value: formato de sequência do valor de atributo de acordo com seu tipo

---

## Capítulo 4. API do IBM Marketing Operations

A API do IBM Marketing Operations é uma fachada que fornece uma visualização de cliente de uma instância em execução do Marketing Operations.

Somente um subconjunto dos recursos do Marketing Operations é mostrado para os usuários. A API é usada simultaneamente por usuários da web do Marketing Operations e por solicitações e acionadores SOAP de Serviço da Web de Serviços de Integração do Marketing Operations. A API suporta os seguintes tipos de operações.

- Criação e exclusão de componentes
- Descoberta (por tipo de componente, valor de atributo e mais valores)
- Inspeção de componente (por meio de seus atributos, links especializados e mais valores)
- Modificação de componentes

---

### Conteúdo da API do IBM Marketing Operations

O pacote `com.unica.publicapi.plan.api` fornece a API do IBM Marketing Operations.

Este pacote oferece interfaces e exceções e contém os seguintes tipos de classes:

- Tipos de dados enumerados.
- Identificadores para identificar instâncias de objeto e de componente.
- Um mapa Java, `AttributeMap`.

A documentação completa da API, incluindo todos os métodos e valores possíveis, está disponível clicando em **Ajuda > Documentação do produto** em uma instância do Marketing Operations e, em seguida, fazendo download do arquivo `IBM <version>PublicAPI.zip`.

### Interfaces da API

A interface de programação de aplicativos (API) do IBM Marketing Operations inclui **IPlanAPI** e **IExecutionContext**.

A API do Marketing Operations inclui as seguintes interfaces.

#### **IPlanAPI**

Define a API pública para o Marketing Operations. Fornece métodos para criar, descobrir e modificar objetos, incluindo pastas, projetos, programas, tarefas de fluxo de trabalho e membros da equipe.

Para sistemas que possuem a integração opcional com o IBM Campaign ativada, também fornece métodos para criar, descobrir e modificar ofertas.

#### **IExecutionContext**

Define os acionadores e bloqueios que executam métodos na API.

### **Métodos de API**

Para obter informações específicas sobre os métodos da API públicos, consulte a classe `iPlanAPI` nos arquivos de documentação da API de JavaDocs.

Esses arquivos estão disponíveis efetuando login no Marketing Operations e selecionando **Ajuda > Documentação do produto** em qualquer página e, em seguida, fazendo download do arquivo `<version>PublicAPI.zip`.

## Exceções comuns

As exceções comuns que são lançadas pela API incluem `NotFoundException`, `AuthorizationException`, `DataException`, `InvalidExecutionContextException` e `NotLockedException`.

A lista a seguir explica por que essas exceções comuns podem ocorrer.

- `<object type>NotFoundException`: O sistema não pode retornar o item ou objeto especificado.
- `AuthorizationException`: O usuário que está associado ao contexto de execução não está autorizado para a operação solicitada. Essa exceção pode ser lançada por qualquer método da API, portanto, é não declarada.
- `DataException`: Ocorreu uma exceção na camada de banco de dados subjacente no IBM Marketing Operations. Verifique o log SQL para obter detalhes.
- `InvalidExecutionContextException`: Há um problema com um contexto de execução transmitido para um método da API (por exemplo, o método não foi inicializado corretamente). Essa exceção pode ser lançada por qualquer API, portanto, é não declarada.
- `NotLockedException`: Tentativa de atualizar dados do componente sem primeiro adquirir o bloqueio necessário. Consulte o método `acquireLock()` da interface `IExecutionContext`.

## Identificadores

Um identificador é um objeto de URL especial que referencia uma determinada instância do objeto em uma instância do IBM Marketing Operations. Os identificadores incluem o tipo de componente, o identificador de dados internos e uma URL base de instância.

Os identificadores usados ou gerados pela API podem ser exteriorizados para uma URL completa. É possível usar a URL resultante de diferentes maneiras. É possível usar a URL para abrir uma visualização do componente na GUI do Marketing Operations, enviá-la em mensagens de email ou usá-la em outro procedimento como um parâmetro.

Os identificadores são válidos apenas para uma determinada instância de serviço ou instância de cluster do Marketing Operations, mas são válidos durante o tempo de vida do serviço implementado. Como resultado, os identificadores podem ser salvos em um arquivo para referência posterior, mas não podem ser usados para acessar componentes em outra instância do Marketing Operations. Essa restrição também se aplica a instâncias no mesmo servidor host físico. No entanto, o Marketing Operations fornece um mecanismo para mapear diferentes URLs base para a instância atual para acomodar a realocação de uma instância em outro servidor (por exemplo, se houver mau funcionamento do equipamento).

Os identificadores são independentes do cliente. Por exemplo, um acionador pode transmitir um identificador para um procedimento, que usa-o como um parâmetro em uma chamada SOAP para um sistema de terceiros. O sistema de terceiros pode então emitir uma solicitação SOAP de volta para o Marketing Operations para iniciar um procedimento que atualiza um atributo.

Os membros da classe de manipulação podem ter factory methods para criar identificadores de vários tipos de URLs. A seguir estão os exemplos.

**Aprovação**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=approvaldetail&approvalid=101`

**Ativo**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=asset&assetMode=VIEW_ASSET&assetid=101`

**Pasta de ativos**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=folder&id=101`

**Biblioteca de ativos**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=library&id=101`

**Anexo**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=attachmentview&attachid=101&parentObjectId=101&parentObjectType=project`

**Conta financeira**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=accountdetails&accountid=101`

**Pasta**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=grouping_folder&folderid=1234`

**Fatura**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=invoicedetails&invoiceid=134`

**Item de linha de fatura**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=invoicedetails&invoiceid=134&line_item_id=101`

**Objeto de marketing**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=componenttabs&componentid=creatives&componentinstid=1234`

**Grade de objeto de marketing**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=componenttabs&componentid=creatives&componentinstid=1234&gridid=grid`

**Linha de grade do objeto de marketing**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=componenttabs&componentid=creatives&componentinstid=1234&gridid=grid&gridrowid=101`

**Equipe de plano**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=teamdetails&func=edit&teamid=100001`

**Usuário de plano**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=adminuserpermissions&func=edit&userId=101`

**Programa**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=programtabs&programid=125`

**Grade do Programa**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=programtabs&programid=1234&gridid=grid`

**Linha de Grade do Programa**

`http://mymachine:7001/plan/affiniumplan.jsp?cat=programtabs&programid=1234&gridid=grid&gridrowid=101`

### Projeto

`http://mymachine:7001/plan/affiniuplan.jsp?cat=projecttabs&projectid=1234`

### Grade do projeto

`http://mymachine:7001/plan/affiniuplan.jsp?cat=projecttabs&projectid=1234&gridid=grid`

### Linha de grade do projeto

`http://mymachine:7001/plan/affiniuplan.jsp?cat=projecttabs&projectid=1234&gridid=grid&gridrowid=101`

### Item de linha do projeto

`http://mymachine:7001/plan/affiniuplan.jsp?cat=projecttabs&projectid=1234&projectlineitemid=123&projectlineitemisversionfinal=false`

### Estágio do fluxo de trabalho

`http://mymachine:7001/plan/affiniuplan.jsp?cat=projectworkflow&projectid=1234&taskid=5678`

### Tarefa de fluxo de trabalho

`http://mymachine:7001/plan/affiniuplan.jsp?cat=projectworkflow&projectid=1234&taskid=5678`

## Mapa de atributos

A classe `AttributeMap` é um mapa Java que contém apenas atributos. O atributo `<Name>` é a chave de entrada de mapa e a matriz de atributo `<values>` (observe o plural) é o valor de entrada de mapa.

A classe `AttributeMap` inclui os seguintes campos.

- `<Name>`: o nome programático do atributo. Este nome serve como uma chave exclusiva para acessar o atributo na instância do componente em que ele ocorre.

**Nota:** `<Name>` não é necessariamente o nome de exibição que é apresentado a um usuário na GUI. Para os componentes que são criados a partir de modelos (como projetos ou tarefas de fluxo de trabalho), o nome do atributo é especificado pela definição de elemento do modelo. O nome do atributo deve ser exclusivo. Para outros componentes, o nome do atributo geralmente é derivado programaticamente da instância do componente do lado do servidor (por exemplo, por meio de introspecção Java).

**Nota:** Por convenção, os atributos customizados incluem o nome do formulário no qual a versão editável é definida: `<form_name>.<attribute_name>`.

- `Values`: uma matriz de objeto Java contendo zero ou mais valores de atributo. O tipo de cada valor deve ser o mesmo e estar de acordo com o tipo do atributo conforme ele está definido no Marketing Operations. Apenas o seguinte wrapper Java e os tipos de Marketing Operations são suportados:
  - `AssetLibraryStateEnum`: um valor de tipo enumerado `AssetLibraryStateEnum`.
  - `AssetStateEnum`: um valor de tipo enumerado `AssetStateEnum`.
  - `AttachmentTypeEnum`: um valor de tipo enumerado `AttachmentTypeEnum`.
  - `AttributeMap`: um mapa que contém atributos.
  - `BudgetPeriodEnum`: um valor de tipo enumerado `BudgetPeriodEnum`.
  - `BudgetTypeEnum`: um valor de tipo enumerado `BudgetTypeEnum`.
  - `Handle`: uma referência a uma instância do componente, linha de grade, atributo, etc.
  - `InvoiceStateEnum`: um valor de tipo enumerado `InvoiceStateEnum`.
  - `java.io.File`: representação de um arquivo.

- java.lang.Boolean: um valor booleano, True ou False
- java.lang.Double: um valor de número decimal de precisão dupla.
- java.lang.Float: um valor de número decimal de precisão única
- java.lang.Integer: um valor de número inteiro de 32 bits
- java.lang.Long: um valor de número inteiro de 64 bits
- java.lang.Object: Objeto Java genérico
- java.lang.String: uma sequência de zero ou mais caracteres Unicode
- java.math.BigDecimal: valor de número decimal assinado de precisão arbitrária. Adequado para moeda; a interpretação do valor depende do código de idioma da moeda para o cliente.
- java.math.BigInteger: valor de número inteiro de precisão arbitrária.
- java.net.URL: um objeto de Universal Resource Locator (URL).
- java.util.ArrayList: Lista de objetos.
- java.util.Calendar: um valor de data e hora para um código de idioma específico.
- java.util.Date: um valor de data e hora. Esse tipo foi descontinuado. Use java.util.Calendar ou java.util.GregorianCalendar no lugar.

**Nota:** Para implementar a data, os usuários podem selecionar java.util.Calendar ou java.util.GregorianCalendar.

- java.util.GregorianCalendar: GregorianCalendar é uma subclasse concreta de java.util.Calendar e fornece o sistema de calendário padrão em uso na maior parte do mundo.
- MonthEnum: um valor de tipo enumerado MonthEnum.
- ProjectStateEnum: um valor de tipo enumerado ProjectStateEnum.
- QuarterEnum: um valor de tipo enumerado QuarterEnum.
- TaskStateEnum: um valor de tipo enumerado TaskStateEnum.
- WeekEnum: um valor de tipo enumerado WeekEnum.

Os metadados de um atributo (como o nome de exibição e descrição traduzidos) são definidos pelo modelo que está associado ao atributo e sua instância de objeto-pai. Os atributos fornecem um mecanismo extensível simples para mostrar os atributos de instância do objeto obrigatórios e opcionais, como nome do projeto, código e data de início.

## Tipos de dados enumerados

A API do IBM Marketing Operations suporta os seguintes tipos de dados e valores enumerados.

### ApprovalMethodEnum

ApprovalMethodEnum define métodos de aprovação válidos. Os valores possíveis são:

- SEQUENTIAL
- SIMULTANEOUS

### ApprovalStateEnum

ApprovalStateEnum define estados de aprovação válidos. Os valores possíveis são:

- CANCELADO
- COMPLETED
- IN\_PROGRESS

- NOT\_STATED
- ON\_HOLD

#### **AssetLibraryStateEnum**

AssetLibraryStateEnum define estados da biblioteca de ativos válidos. Os valores possíveis são:

- DISABLED
- ENABLED

#### **AssetStateEnum**

AssetStateEnum define os estados de ativos válidos. Os valores possíveis são:

- ARCHIVE
- DRAFT
- FINALIZE
- LOCK

#### **AttachmentTypeEnum**

AttachmentTypeEnum define tipos de anexo válidos. Os valores possíveis são:

- ASSET
- FILE
- URL

#### **BudgetPeriodEnum**

BudgetPeriodEnum define os possíveis períodos de orçamento. Os valores possíveis são:

- ALL
- MONTHLY
- QUARTERLY
- WEEKLY
- YEARLY

#### **BudgetTypeEnum**

BudgetTypeEnum define tipos de orçamento válidos. Os valores possíveis são:

- ACTUAL
- ALLOCATED
- COMMITTED
- FORECAST
- TOTAL

#### **ComponentTypeEnum**

ComponentTypeEnum identifica os tipos de componentes acessíveis do Marketing Operations. Os valores possíveis são:

- APPROVAL
- ASSET
- ASSET\_FOLDER
- ASSET\_LIBRARY
- ATTACHMENT
- FINANCIAL\_ACCOUNT
- GROUPING\_FOLDER

- INVOICE
- MARKETING\_OBJECT
- PLAN\_TEAM
- PLAN\_USER
- PROGRAM
- PROJECT
- PROJECT\_REQUEST
- TASK
- 

#### **InvoiceStateEnum**

InvoiceStateEnum define estados de fatura válidos. Os valores possíveis são:

- CANCELADO
- DRAFT
- PAID
- PAYABLE

#### **MonthEnum**

MonthEnum define valores válidos para o mês.

#### **OfferStateEnum**

OfferStateEnum define estados de oferta válidos. Os valores possíveis são:

- STATE\_OFFER\_DRAFT
- STATE\_OFFER\_PUBLISHED
- STATE\_OFFER\_RETIRED

#### **ProjectCopyTypeEnum**

ProjectCopyTypeEnum define métodos válidos para a cópia de um projeto. Os valores possíveis são:

- COPY\_USING\_PROJECT\_METRICS
- COPY\_USING\_TEMPLATE\_METRICS

#### **ProjectParticipantLevelEnum**

ProjectParticipantLevelEnum identifica as funções que os usuários podem ter em um projeto. Os valores possíveis são:

- OWNER
- PARTICIPANT
- REQUESTER

#### **ProjectStateEnum**

ProjectStateEnum define estados válidos de projeto e de solicitação. Os valores possíveis são:

- ACCEPTED
- CANCELADO
- COMPLETED
- DRAFT
- IN\_PROGRESS
- IN\_RECONCILIATION
- LATE: o projeto não foi iniciado por sua data de início planejada.
- NOT\_STARTED
- ON\_HOLD

- OVERDUE: o projeto não foi concluído antes de sua data de encerramento planejada.
- RETURNED
- SUBMITTED

Para obter informações adicionais sobre status de projeto e de tarefa, consulte o *Guia do Usuário do IBM Marketing Operations*.

**QuarterEnum**

QuarterEnum define os valores válidos para trimestres: Q1, Q2, Q3 e Q4.

**TaskStateEnum**

TaskStateEnum define estados válidos da tarefa de fluxo de trabalho. Os valores possíveis são:

- ACTIVE
- DISABLED
- FINISHED
- PENDING
- SKIPPED

**WeekEnum**

WeekEnum define valores válidos para semanas em um ano, de WEEK\_1 a WEEK\_53.

---

## Antes de entrar em contato com o suporte técnico da 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. Use 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 da IBM para obter mais informações.

**Nota:** O Suporte Técnico não grava ou cria scripts de API. Para obter assistência na implementação de nossas ofertas de API, entre em contato com o IBM Professional Services.

### Informações a serem reunidas

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

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

### Informações do 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 um arquivo `version.txt` que está localizado sob o diretório de instalação para seu aplicativo.

### Informações de contato para o suporte técnico da IBM

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

**Nota:** Para inserir uma solicitação de suporte, você deve efetuar login com uma conta 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 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 um representante IBM local para obter informações sobre produtos e serviços disponíveis atualmente em sua área. Qualquer referência a produtos, programas ou serviços IBM não significa que apenas produtos, programas ou serviços IBM possam ser usados. Qualquer produto, programa ou serviço funcionalmente equivalente, que não infrinja nenhum direito de propriedade intelectual da IBM poderá ser usado em substituição a este produto, programa ou serviço da IBM. Entretanto, a avaliação e verificação da operação de qualquer produto, programa ou serviço não IBM são de responsabilidade do usuário.

A IBM pode ter patentes ou solicitações de patentes pendentes relativas a assuntos tratados nesta publicação. O fornecimento desta publicação não lhe garante direito algum sobre tais patentes. Pedidos de licença devem 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 pedidos de licença relacionados a informações de DBCS (Conjunto de Caracteres de Byte Duplo), entre em contato com o Departamento de Propriedade Intelectual da IBM em seu país ou envie pedidos de licença, por escrito, para:

IBM World Trade Asia Corporation Licensing  
2-31 Roppongi 3-chome,  
Minato-ku  
Tokyo 106,  
Japan

O parágrafo a seguir não se aplica ao Reino Unido nem 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 OU CONDIÇÕES 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 mudanças periódicas nas informações aqui contidas; tais mudanças serão incorporadas em futuras edições desta publicação. A IBM pode, a qualquer momento, aperfeiçoar e/ou alterar os produtos e/ou programas descritos nesta publicação, sem aviso prévio.

As referências nestas informações a websites não IBM são fornecidas apenas por conveniência e não representam de forma alguma um endosso a esses websites. Os materiais contidos nesses websites não fazem parte dos materiais desse produto IBM e o uso desses websites é de inteira responsabilidade do Cliente.

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

Os licenciados deste programa que desejarem obter informações sobre este assunto com o propósito de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) o uso mútuo das informações trocadas, deverão entrar em contato com:

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

Tais informações podem estar disponíveis, sujeitas aos 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 estas medidas serão iguais em sistemas geralmente disponíveis. Além disso, algumas medidas podem ter sido estimadas por extrapolação. Os resultados reais podem variar. Os usuários deste documento devem verificar os dados aplicáveis para seu ambiente específico.

As informações relativas a produtos não IBM foram obtidas junto aos fornecedores dos respectivos produtos, a partir de seus anúncios publicados ou de outras fontes disponíveis publicamente. A IBM não testou estes produtos e não pode confirmar a precisão de seu desempenho, compatibilidade nem qualquer outra reivindicação relacionada a produtos não IBM. As dúvidas sobre os recursos de produtos não IBM devem ser encaminhadas diretamente aos 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 IBM mostrados são preços de varejo sugeridos pela IBM, são atuais e estão sujeitos a mudança sem aviso prévio. Os preços do revendedor podem variar.

Estas informações contêm exemplos de dados e relatórios usados nas operações diárias de negócios. Para ilustrá-los da forma mais completa possível, os exemplos podem incluir nomes de indivíduos, empresas, marcas e produtos. Todos estes nomes são fictícios e qualquer semelhança com nomes e endereços usados por uma empresa real é mera coincidência.

## LICENÇA DE COPYRIGHT:

Estas informações contêm programas de aplicativos de amostra na linguagem fonte, ilustrando as técnicas de programação em diversas plataformas operacionais. O Cliente pode copiar, modificar e distribuir estes programas de amostra sem a necessidade de pagar à IBM, com objetivos de desenvolvimento, uso, marketing ou distribuição de programas de aplicativos em conformidade com a interface de programação de aplicativo para a plataforma operacional para a qual os programas de amostra são criados. Esses exemplos não foram testados completamente em todas as condições. Portanto, a IBM não pode garantir ou implicar a 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 responsabiliza por danos causados pelo uso dos programas de amostra.

Se estiver visualizando essas informações em formato eletrônico, as fotografias e ilustrações coloridas podem não aparecer.

---

## Marcas Comerciais

IBM, o logotipo IBM e [ibm.com](http://ibm.com) são marcas comerciais ou marcas registradas da International Business Machines Corp., registradas em vários países no mundo todo. Outros nomes de produtos e serviços 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](http://www.ibm.com/legal/copytrade.shtml).

---

## Política de Privacidade e Termos de Considerações de Uso

Os produtos de Software IBM, incluindo software como soluções de 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, customizar interações com o usuário final ou para outros propósitos. Um cookie é uma parte de dados que um website pode enviar ao seu navegador, que, em seguida, pode ser armazenado em seu computador como uma tag que identifica o computador. Em muitos casos, nenhuma informação pessoal é coletada por esses cookies. Se uma Oferta de Software, que esteja sendo usada, permitir que você colete informações pessoais através de cookies e tecnologias semelhantes, nós o informaremos sobre as especificações abaixo.

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

Várias jurisdições regulam a coleta de informações pessoais por meio de cookies e tecnologias semelhantes. Se as configurações implementadas para esta Oferta de Software fornecerem a você, como cliente, a capacidade de coletar informações pessoais de usuários finais por meio de cookies e outras tecnologias, é necessário procurar o seu próprio conselho jurídico sobre todas as leis aplicáveis a essa coleta de dados, incluindo quaisquer requisitos para o fornecimento de aviso e consentimento quando apropriado.

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 fins, consulte Declaração de Privacidade Online da IBM em <http://www.ibm.com/privacy/details/us/en> na seção autorizada "Cookies, Web Beacons e Outras Tecnologias."





Impresso no Brasil