

Versione 10 Release 1
17 novembre 2017

*IBM Marketing Operations Modulo di
integrazione*



Nota

Prima di utilizzare queste informazioni ed il prodotto che supportano, leggere le informazioni contenute in "Informazioni particolari" a pagina 33.

Questa edizione si applica alla versione 10, release 1, modifica 0 di IBM Marketing Operations ed a tutte le modifiche e release successive, se non diversamente indicato nelle nuove edizioni.

© Copyright IBM Corporation 2002, 2017.

Indice

Capitolo 1. Definizione dei servizi di integrazione di IBM Marketing

Operations 1

Requisiti per i servizi di integrazione di Marketing Operations 2

Elementi di base dei servizi di integrazione di IBM Marketing Operations 3

 Installazione dei servizi di integrazione 5

 Contenuto dell'SDK (Software Developer Kit) 5

JavaDocs ospitate 6

Documentazione e guida di Marketing Operations. . . 6

Capitolo 2. Servizio web di integrazione di Marketing Operations. 9

WSDL dei servizi di integrazione di Marketing

Operations 9

executeProcedures 9

Tipi di dati del servizio web di integrazione

Marketing Operations 10

Capitolo 3. Procedure di IBM Marketing Operations 15

Premesse 15

Parametri di configurazione 17

Progettazione. 17

Ciclo di vita della procedura 18

Classi Java chiave 19

Blocco dei dati 20

Transazioni della procedura 20

Comunicazione della procedura 20

Registrazione della procedura 21

File di definizione del plugin della procedura . . . 21

Capitolo 4. API SOAP IBM Marketing Operations 23

Contenuto dell'API SOAP IBM Marketing

Operations 23

 Interfacce API SOAP 23

 Eccezioni comuni API SOAP. 24

 Handle API SOAP 24

 AttributeMap API SOAP 26

 Tipi di dati elencati API SOAP 27

Prima di contattare il supporto tecnico di IBM 31

Informazioni particolari. 33

Marchi 35

Normativa sulla privacy e termini di utilizzo -

Considerazioni 35

Capitolo 1. Definizione dei servizi di integrazione di IBM Marketing Operations

I servizi di integrazione di IBM® Marketing Operations combinano il servizio web di integrazione di Marketing Operations, le procedure API SOAP e i trigger alle funzioni di business estese.

I servizi di integrazione di IBM Marketing Operations sono composti dai seguenti componenti.

- **Servizio web di integrazione di Marketing Operations**

I servizi di integrazione consentono ai clienti di Marketing Operations e ai servizi professionali di IBM di integrare Marketing Operations con altre applicazioni eseguite nell'ambiente.

- **Procedure e API SOAP Marketing Operations**

In Marketing Operations è possibile definire delle procedure personalizzate per estendere la logica di business di Marketing Operations in modi arbitrari. Dopo aver definito le procedure, queste possono essere la destinazione delle chiamate del servizio web dei servizi di integrazione da altre applicazioni. Le procedure possono essere definite anche per inviare i messaggi ad altre applicazioni.

- **Marketing Operations trigger**

I trigger possono essere associati a eventi e procedure in Marketing Operations. Quando si verifica un evento, viene eseguito il trigger associato.

Le API REST non utilizzano i servizi di integrazione di Marketing Operations. Per informazioni sulle API REST, consultare IBM Marketing Operations - Guida dell'amministratore.

Versioni e compatibilità con le versioni precedenti

Le versioni future dei servizi di integrazione saranno compatibili con le versioni precedenti con tutte le release minori e di manutenzione che condividono un numero di versione maggiore. Tuttavia, IBM si riserva il diritto di interrompere la compatibilità con una versione precedente per le release punto zero (x.0) per motivi tecnici o aziendali.

Il numero di versione maggiore di questa API viene incrementato se si apporta una qualsiasi di queste modifiche.

- Modifiche dell'interpretazione dei dati
- Modifiche della logica di business (ad esempio, se cambiano le funzioni del metodo di servizio)
- Cambiano i parametri del metodo, i tipi restituiti o entrambi

Il numero della versione minore dell'API viene incrementato se viene apportata una qualsiasi delle seguenti modifiche. Queste modifiche sono compatibili con una versione precedente per definizione.

- Viene aggiunto un nuovo metodo
- Viene aggiunto un nuovo tipo di dati e il suo utilizzo viene limitato a un nuovo metodo
- Viene aggiunto un nuovo elemento a un tipo elencato

- Viene definita una nuova versione di un'interfaccia con un suffisso della versione

Autenticazione

L'autenticazione non è obbligatoria; tutti i client sono associati a un utente noto di IBM Marketing Operations, denominato PlanAPIUser. Un amministratore di sistema configura le funzioni di sicurezza di questo utente speciale per soddisfare le esigenze di tutti i client del servizio web.

Locale

L'unica locale supportata è quella attualmente configurata per l'istanza del sistema IBM Marketing Operations. Tutti i dati dipendenti dalla locale, ad esempio i messaggi e la valuta, si suppone siano nella locale del sistema.

Gestione dello stato

L'API e il servizio web sono *senza stato*; nessuna informazione per client viene salvata dall'implementazione del servizio mediante le chiamate API. Questa funzione offre un'implementazione del servizio efficiente e semplifica il supporto del cluster.

Transazioni del database

I servizi di integrazione di Marketing Operations non mostrano le transazioni del database al client, ma utilizzano tali informazioni se sono incluse nel contesto di esecuzione. Se viene avviata una transazione, l'effetto di tutte le chiamate API in una particolare procedura è atomico. In altre parole, una chiamata API non riuscita lascia il database nello stesso stato in cui si troverebbe se l'API non fosse stata chiamata. Gli altri utenti di Marketing Operations non vedono le modifiche finché la procedura non completa la transazione.

Le chiamate API che aggiornano il database devono prima acquisire un blocco di modifica per impedire che altri utenti modifichino i dati sottostanti durante le chiamate API. Gli altri utenti non possono aggiornare i componenti bloccati finché non viene completata la chiamata API. Allo stesso modo, l'utente successivo di Marketing Operations o il client API devono acquisire il blocco sui dati prima che venga inoltrata un'altra chiamata API.

Elaborazione eventi

Le operazioni sui componenti di IBM Marketing Operations mediante l'API generano gli stessi eventi che vengono generati quando un utente di Marketing Operations esegue l'operazione. Gli utenti che eseguono la sottoscrizione a determinate notifiche, ad esempio, alle modifiche di stato del progetto, vengono informati delle modifiche di stato che risultano dalle chiamate API e dalle azioni dell'utente.

Requisiti per i servizi di integrazione di Marketing Operations

I servizi di integrazione di Marketing Operations hanno i seguenti requisiti.

I servizi di integrazione di Marketing Operations devono:

- Essere agganciati all'integrazione di sistema in modo libero.

- Fornire un meccanismo per le applicazioni del cliente per influire su Marketing Operations mediante le chiamate del servizio web.
- Fornire un meccanismo per le applicazioni del cliente per essere informati di alcuni eventi in Marketing Operations.
- Fornire un modello di programmazione semplice, di facile comprensione e utilizzo.
- Essere solido quando viene ripristinato dopo un errore.
- Garantire l'integrità dei dati.
- Integrarsi e ridurre l'effetto sui clienti esistenti di Marketing Operations basati sulla GUI.
- Fornire accesso specifico ai componenti di Marketing Operations, non coinvolgendo i programmatori nei dettagli di implementazione sottostanti.

Elementi di base dei servizi di integrazione di IBM Marketing Operations

Utilizzare i servizi di integrazione di IBM Marketing Operations per creare le procedure personalizzate. È possibile utilizzare queste procedure per attivare gli eventi esterni quando si verificano determinati eventi in Marketing Operations. È possibile utilizzare queste procedure per eseguire le funzioni di Marketing Operations da sistemi o programmi esterni.

L'interfaccia API interagisce con IBM Marketing Operations a livello programmatico, così come la GUI si interfaccia con Marketing Operations a livello di utente. Utilizzando l'API, si costruiscono le procedure. Utilizzando queste procedure, è possibile la comunicazione tra Marketing Operations e i sistemi esterni. Il servizio web di Marketing Operations è il contenitore di procedure, API e trigger.

L'architettura dei servizi di integrazione di Marketing Operations viene mostrata di seguito.

Di seguito vengono indicati i componenti chiave dei servizi di integrazione.

- Marketing Operations Procedure Manager: estende la logica di business interagendo con Marketing Operations mediante l'API.
- Marketing Operations Trigger Manager: associa una condizione (ad esempio, la modifica di stato di un oggetto di marketing) a un'azione (una procedura da eseguire quando viene soddisfatta la condizione per il trigger).

Metodi

Utilizzare i componenti dei servizi di integrazione di IBM Marketing Operations per sviluppare procedure personalizzate, come mostrato nel seguente diagramma.

Dopo avere installato SDK (Software Developer Kit), effettuare gli step di base riportati di seguito.

1. Codificare la procedura personalizzata.
2. Aggiornare la definizione del plugin nel file di definizione XML.
3. Creare il plugin nel seguente modo.
 - a. Compilare le classi necessarie.

- b. Se si utilizza una libreria di terze parti non contenuta nell'archivio di Marketing Operations, inserire la libreria nel file `plan.war` e rieseguire la distribuzione.
4. Riavviare Marketing Operations. Le modifiche alle classi della procedura vengono applicate quando si riavvia il server delle applicazioni.

Nota: se si modifica il file `plan.war`, è necessario rimuovere la distribuzione e rieseguire la distribuzione di Marketing Operations con il nuovo file `plan.war`. Rimuovere la distribuzione e rieseguire la distribuzione di Marketing Operations se si utilizza una libreria di terze parti non inclusa nell'archivio di Marketing Operations e si modifica il file `plan.war`.

Esempio di base di comunicazioni tra IBM Marketing Operations e l'API

Nel seguente esempio di base viene descritto come stabilire le comunicazioni tra l'API e Marketing Operations. Non esegue alcun lavoro utile; esegue un round trip tra Marketing Operations e i servizi di integrazione.

Questo esempio utilizza parti delle procedure di esempio incluse nell'SDK (Software Developer Kit) dei servizi di integrazione di Marketing Operations. Specificamente, è possibile trovare il codice indicato qui nei seguenti file.

- `PlanClientFacade.java`
- `PlanWSNOOPTestCase.java`

Il metodo `noop` è una chiamata del servizio web a Marketing Operations. È definito nella classe `PlanClientFacade` e passa valori null in un array.

```
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);
}
```

La procedura `testExecuteProcedure` chiama il metodo `noop` da `PlanClientFacade` per stabilire un round trip con l'applicazione Marketing Operations.

```
public void testExecuteProcedure() throws Exception {
    // Time out after a minute
    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;

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

Per i dettagli di `NameValueArrays`, `ProcedureResponse` e altri metodi e tipi di dati elencati, fare riferimento a *Marketing Operations Integration Module* e a *JavaDocs*.

Installazione dei servizi di integrazione

Il modulo servizi di integrazione di IBM Marketing Operations è un componente separato a pagamento. Se si acquista il modulo servizi di integrazione, è necessario installarlo.

1. Eseguire il download dei programmi di installazione dei servizi di integrazione di IBM Marketing Operations.
2. I programmi di installazione di IBM Marketing Software rilevano il modulo servizi di integrazione.
3. Il programma di installazione imposta le proprietà di configurazione in **Marketing Operations | `umoConfiguration | integrationServices | enableIntegrationServices`**. È possibile personalizzare l'installazione modificando i parametri di configurazione. Per ulteriori informazioni, consultare "Parametri di configurazione" a pagina 17.

Contenuto dell'SDK (Software Developer Kit)

L'SDK (Software Developer Kit) contiene la documentazione contenente tutte le interfacce e le classi pubblicapi e il codice di esempio.

Per le API SOAP, tutti i componenti dei servizi di integrazione di Marketing Operations vengono installati in una cartella denominata `devkits`.

Il codice di esempio viene installato nelle seguenti cartelle.

- La cartella **build** contiene gli script per creare e distribuire le procedure personalizzate.
- La cartella **Classes** contiene le classi della procedura compilate.
Gli utenti devono distribuire le classi compilate delle loro procedure personalizzate nel percorso specificato dal parametro di configurazione **integrationProcedureClasspathURL**. Quindi, il gestore della procedura IBM Marketing Operations le carica nel modo specificato nel file di configurazione `procedure-plugins.xml`.
- La cartella **lib** contiene le librerie necessarie per sviluppare e compilare le procedure personalizzate.
- La cartella **src** contiene i file di origine per le procedure personalizzate. Gli utenti possono inserire qui le procedure personalizzate da avviare come trigger o servizi web. Solo le API SOAP supportano le procedure personalizzate.
 - La cartella **src/procedure** contiene il file di configurazione `procedure-plugins.xml`. Ogni procedura personalizzata che viene eseguita come un trigger in base a un evento o mediante un servizio web esterno, deve avere una voce in questo file. Le voci devono contenere un percorso classe completo della procedura e i parametri di inizializzazione richiesti.
 - La cartella **src/procedure** contiene anche alcune procedure di esempio, che sono incluse con IBM Marketing Operations. Queste procedure possono essere utilizzate per comprendere e sviluppare le procedure personalizzate.
Inserire le procedure personalizzate nella directory **src** in una nuova struttura di cartelle, ad esempio, `com/<mycompany>/<mypackage>`. Non inserire le procedure personalizzate nella cartella delle procedure di esempio.
 - La cartella **src/soap** contiene i client del servizio web di esempio, che sono sviluppati in Java. Utilizzare questi esempi come un punto di partenza per sviluppare i client del servizio web per i servizi di integrazione. Questa cartella contiene anche gli script binari per avviare i client di esempio nella riga di comando.

JavaDocs ospitate

Per informazioni specifiche sui metodi dell'API pubblica, fare riferimento alla classe iPlanAPI nei file della documentazione dell'API JavaDocs.

Questi file sono disponibili nei seguenti modi.

- Dai file nella directory `<IBM_IMS>/<MarketingOperations_Home>/devkits/integration/javadocs` per l'API SOAP sul server che ospita Marketing Operations.
- Eseguendo l'accesso a Marketing Operations e selezionando **Guida > Documentazione prodotto** da qualsiasi pagina e scaricando il file `IBM <version>PublicAPI.zip` per l'API SOAP.

Documentazione e guida di Marketing Operations

Persone diverse all'interno della propria organizzazione utilizzano IBM Marketing Operations per svolgere attività differenti. Informazioni su Marketing Operations sono disponibili in un insieme di guide, ciascuna delle quali è rivolta ai membri del team con conoscenze ed obiettivi specifici.

La tabella seguente descrive le informazioni disponibili in ciascuna guida.

Tabella 1. Guide nella serie della documentazione Marketing Operations.

La tabella a tre colonne riportata di seguito descrive le attività in una colonna, i nomi dei manuali nella seconda e il destinatario nella terza.

Se l'utente	Consultare	Destinatario
• Pianifica e gestisce progetti • Stabilisce attività del flusso di lavoro, eventi chiave e personale • Tiene traccia delle spese del progetto • Ottiene revisioni ed approvazioni del contenuto • Produce report	<i>IBM Marketing Operations Guida dell'utente</i>	• Project manager • Designer creativi • Gestori del marketing tramite pubblicità diretta per posta
• Progettano modelli, moduli, attributi e metriche • Personalizzano l'interfaccia utente • Definiscono la sicurezza ed il livello di accesso dell'utente • Implementano funzioni facoltative • Configurano e regolano Marketing Operations	<i>IBM Marketing Operations Guida per l'amministratore</i>	• Project manager • Amministratori IT • Consulenti dell'implementazione
• Creano campagne di marketing • Pianificano le offerte • Implementano l'integrazione tra Marketing Operations e Campaign • Implementano l'integrazione tra Marketing Operations e IBM Digital Recommendations	<i>IBM Marketing Operations e IBM Guida di integrazione</i>	• Project manager • Specialisti di esecuzione del marketing • Gestori del marketing diretto

Tabella 1. Guide nella serie della documentazione Marketing Operations (Continua).

La tabella a tre colonne riportata di seguito descrive le attività in una colonna, i nomi dei manuali nella seconda e il destinatario nella terza.

Se l'utente	Consultare	Destinatario
• Desidera ottenere informazioni sulle nuove funzioni del sistema • Ricercano problemi noti e soluzioni temporanee	<i>IBM Marketing Operations Release Notes</i>	Chiunque utilizzi Marketing Operations
• Installare Marketing Operations • Configurare Marketing Operations • Aggiornare alla nuova versione di Marketing Operations	<i>IBM Marketing Operations Guida all'installazione</i>	• Consulenti di implementazione software • Amministratori IT • Amministratori del database
Creare procedure personalizzate per integrare Marketing Operations con altre applicazioni.	<i>IBM Marketing Operations Modulo di integrazione e API JavaDoc disponibili quando si fa clic su Guida > Documentazione del prodotto in Marketing Operations, e poi si scarica il file IBM<version>PublicAPI.zip per l'API SOAP e IBM<version>PublicAPI-RestClient.zip per l'API REST.</i>	• Amministratori IT • Amministratori del database • Consulenti dell'implementazione
Apprendere la struttura del database Marketing Operations	<i>Schema del sistema IBM Marketing Operations</i>	Amministratori del database
Ulteriori informazioni mentre si lavora	• Richiamare la guida ed eseguire una ricerca oppure sfogliare <i>Marketing Operations - Guida dell'utente, Guida dell'amministratore</i> oppure <i>Marketing Operations - Guida all'installazione</i>: Fare clic su Guida > Guida per questa pagina • Accedere a tutte le guide Marketing Operations: Guida > Documentazione del prodotto • Accedere alle guide per tutti i prodotti IBM Marketing Software: Fare clic su Guida > Tutto IBM Marketing Software Tutta la documentazione	Chiunque utilizzi Marketing Operations

Capitolo 2. Servizio web di integrazione di Marketing Operations

Il servizio web offre una vista client dei servizi di integrazione di Marketing Operations ed è parte della distribuzione del server IBM Marketing Operations. Il servizio viene utilizzato simultaneamente con gli utenti del web di Marketing Operations.

Il servizio web supporta una chiamata API, `executeProcedure`.

Un client effettua direttamente questa chiamata del servizio web.

WSDL dei servizi di integrazione di Marketing Operations

La WSDL (Web Services Definition Language) è stata definita manualmente ed è la parola finale sulla definizione del servizio web.

Axis

Questa versione del servizio web utilizza Axis2 1.5.2 per generare le classi lato server che costituiscono l'implementazione del servizio web dal file WSDL. Gli utenti possono utilizzare qualsiasi versione di Axis, o una tecnica non Axis, per creare un'implementazione lato client per l'integrazione con l'API dalla WSDL fornita.

Versione del protocollo

La versione del protocollo è esplicitamente collegata alla WSDL nel seguente modo:

- Come parte del nome WSDL, ad esempio, `PlanIntegrationService1.0.wsdl`
- Come parte del targetNamespace WSDL, ad esempio, `xmlns:tns="http://webservices.unica.com /MktOps/services/PlanIntegrationServices1.0?wsdl"`

WSDL

Un file WSDL viene fornito con i servizi di integrazione di IBM Marketing Operations: `PlanIntegrationServices1.0.wsdl`. La WSDL è inserita nella directory `integration/examples/soap/plan`. Lo script di creazione di esempio utilizza questo file per generare gli stub lato client appropriati per la connessione al servizio web.

`executeProcedure`

`executeProcedure` è la chiamata API supportata dal servizio web.

Sintassi

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

Valori restituiti

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

Descrizione

Questo metodo richiama la procedura specificata con un array di parametri facoltativo. La chiamata viene eseguita in modo sincrono; ovvero, blocca il client e restituisce il risultato dopo il completamento.

Parametri

Tabella 2. Parametri di *executeProcedure*

Nome	Descrizione
key	La chiave unica della procedura da eseguire. Viene visualizzato un errore <i>RemoteException</i> se nessuna procedura è collegata a key .
jobid	Stringa facoltativa che identifica il job associato a questa esecuzione della procedura. Questa stringa è un elemento pass-through, ma può essere utilizzato per collegare i job del client all'esecuzione di una determinata procedura.
paramArray	Un array di parametri da passare alla procedura. Vengono visualizzati un messaggio e uno stato di errore se uno o più dei parametri non sono validi (ad esempio, tipo errato o valore non corretto). Il client determinerà i parametri, il loro tipo e il numero di valori richiesti dalla procedura.

Parametri restituiti

Tabella 3. Parametri restituiti da *executeProcedure*

Nome	Descrizione
status	Un codice formato da numeri interi: • 0 indica che la procedura è stata eseguita correttamente • un numero intero indica un errore Le procedure possono utilizzare lo stato per indicare livelli di errori differenti.
messages	Un array di zero o più strutture di dati di messaggi. Se status è 0, questo array non contiene messaggi di errore, ma potrebbe contenere messaggi informativi o di avviso. Se status è diverso da zero, i messaggi potrebbero contenere una combinazione di messaggi di errore, informativi e di avviso.

Tipi di dati del servizio web di integrazione Marketing Operations

I tipi di dati utilizzati dal servizio web sono indipendenti da un determinato bind di servizio o implementazione di programmazione.

Viene utilizzata la seguente notazione.

- *<type>*: *<type definition>* definisce un tipo di dati semplice. Ad esempio:
Handle: string
- *<type>*: [*<type definition>*] definisce un tipo di dati complesso o una struttura di dati complessa.
- *<type>*: { *<type definition>* } definisce un tipo di dati complesso o una struttura di dati complessa.

Gli elementi di tipo complesso e i parametri API possono utilizzare questi tipi per dichiarare gli array. Ad esempio:

```
Handle [] handles
```

Il tipo, handles, è un array di tipi Handle.

Tipi primitivi

I tipi primitivi sono limitati ai tipi definiti nella seguente tabella per semplificare il supporto per i bind SOAP 1.1. Tutti i tipi possono essere dichiarati come array, ad esempio, **String** []. In modo intrinseco, i tipi di dati binari, ad esempio **long**, possono essere rappresentati come stringhe da un bind del protocollo (ad esempio, SOAP). Questa rappresentazione, tuttavia, non ha alcun effetto sulla semantica del tipo e sui valori consentiti visti dal client.

Tabella 4. Tipi primitivi

Tipo API	Descrizione	Tipo SOAP	Tipo Java™
Boolean	Valore booleano: true o false	xsd:Boolean	Boolean
dateTime	Un valore data/ora	xsd:datetime	Date
decimal	Un valore decimale a precisione arbitraria	xsd:decimal	java.math.BigDecimal
double	Un valore decimale con segno, a precisione doppia	xsd:double	double
int	Un valore intero con segno, a 32 bit	xsd:int	int
integer	Un valore intero con segno, a precisione arbitraria	xsd:integer	java.math.BigInteger
long	Un valore intero con segno, a 64 bit	xsd:long	long
string	Una stringa di caratteri Unicode	xsd:string	java.lang.String

MessageTypeEnum

```
MessageTypeEnum: { INFORMATION, WARNING, ERROR }
```

MessageTypeEnum è un tipo elencato che definisce tutti i possibili tipi di messaggio.

- INFORMATION: un messaggio informativo
- WARNING: un messaggio di avviso
- ERROR: un messaggio di errore

Messaggio

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

Il messaggio è una struttura di dati che definisce il risultato di una chiamata API del servizio web. Fornisce i campi facoltativi per un codice non localizzato, un testo localizzato e i dettagli del log. Attualmente, tutto il testo localizzato utilizza la locale impostata per l'istanza del server IBM Marketing Operations.

Tabella 5. Parametri del messaggio

Parametro	Descrizione
type	Un MessageTypeEnum, che imposta il tipo di messaggio.
code	Un codice facoltativo, in formato stringa, per il messaggio.
localizedText	Una stringa di testo facoltativa da associare al messaggio.
logDetail	Un messaggio della traccia di stack facoltativo.

NameValue

NameValue: [string name, int sequence]

NameValue è un tipo complesso di base che definisce la coppia nome-valore. Inoltre, definisce una sequenza facoltativa utilizzata dal servizio per creare gli array di valori necessari (le sequenze in base zero).

Tutti i NameValue con lo stesso nome, ma con numeri di sequenza differenti, vengono convertiti in un array di valori e vengono associati al nome comune.

La dimensione dell'array è determinata dal numero di sequenza massimo; gli elementi dell'array non specificati hanno valori null. I numeri di sequenza dell'array devono essere univoci. Il valore e il relativo tipo vengono forniti dal tipo esteso.

Tabella 6. Parametri di NameValue

Parametro	Descrizione
name	Una stringa che definisce il nome di un tipo NameValue.
sequence	Un numero intero in base zero, che imposta il numero di sequenza per il valore implicito NameValue.

I tipi estesi NameValue sono definiti per ogni tipo primitivo nel seguente modo.

Tabella 7. Tipi estesi NameValue

Tipo esteso	Descrizione
BigDecimalNameValue: NameValue [decimal value]	Un tipo NameValue il cui valore è un numero decimale a precisione arbitraria.
BigIntegerNameValue: NameValue [integer value]	Un tipo NameValue il cui valore è un numero intero di dimensione arbitraria.
BooleanNameValue: NameValue [Boolean value]	Un tipo NameValue il cui valore è booleano.
CurrencyNameValue: NameValue [string locale, decimal value]	Un tipo NameValue adatto per rappresentare la valuta in alcune locale. La locale è un codice lingua ISO, ovvero un codice di due lettere minuscole definito da ISO-639. Attualmente, la locale deve essere conforme alla locale impostata nell'istanza del server IBM Marketing Operations.
DateNameValue: NameValue [datetime value]	Un tipo NameValue il cui valore è una data.
DecimalNameValue: NameValue [double value]	Un tipo NameValue il cui valore è un numero decimale a precisione doppia.

Tabella 7. Tipi estesi NameValue (Continua)

Tipo esteso	Descrizione
IntegerNameValue: NameValue [long value]	Un tipo NameValue il cui valore è un numero intero a 64 bit.
String NameValue: NameValue [string value]	Un tipo NameValue il cui valore è una stringa.

Infine, è definito un array di tipi estesi NameValue per essere utilizzato quando bisogna definire una serie di NameValue di tipi differenti.

```

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

```

Capitolo 3. Procedure di IBM Marketing Operations

Una "procedura" è una classe Java personalizzata o standard ospitata da IBM Marketing Operations, che esegue alcune unità di lavoro. Le procedure offrono un modo ai clienti e ai servizi finanziari di IBM di estendere la logica di business di Marketing Operations in modi arbitrari.

Le procedure seguono un modello di programmazione semplice con un'API ben definita per influire sui componenti gestiti da Marketing Operations. Le procedure vengono "rilevate" mediante un meccanismo di ricerca semplice e un file di definizione basato su XML. Marketing Operations esegue le procedure in base alle esigenze dei "client". Ad esempio, in risposta a una richiesta di integrazione (in entrata) o all'attivazione di un trigger (interno o in uscita).

Le procedure vengono eseguite in modo sincrono con il client; i risultati vengono resi disponibili direttamente al client e mediante un meccanismo di controllo persistente. L'esecuzione di una procedura può anche causare l'attivazione di altri eventi e trigger in Marketing Operations.

Le procedure devono essere scritte in Java.

Premesse

Le classi di implementazione della procedura vengono fornite in un file JAR o in una struttura ad albero di classi separata e resa disponibile a IBM Marketing Operations mediante un percorso dell'URL.

Implementazione della procedura

Il gestore dell'esecuzione della procedura utilizza un programma di caricamento classe indipendente per caricare queste classi. Per impostazione predefinita, Marketing Operations cerca nella seguente directory.

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

Per modificare questa impostazione predefinita, impostare il parametro **integrationProcedureClasspathURL** in **Impostazioni > Configurazione > Marketing Operations > umoConfiguration > integrationServices**.

Il nome della classe di implementazione della procedura segue le convenzioni di denominazione Java accettate per evitare conflitti del package con "unica" e le classi di altri fornitori. In particolare, i clienti non devono inserire le procedure nella struttura ad albero di package "com.unica" o "com.unicacorp".

L'implementazione della procedura è codificata nella versione di runtime Java utilizzata da IBM Marketing Operations sul server delle applicazioni (almeno JRE 1.5.10).

La classe di implementazione della procedura viene caricata dalla politica di caricamento classe che viene utilizzata normalmente da IBM Marketing Operations (in genere **le classi padre per ultime**). Il server delle applicazioni potrebbe fornire gli strumenti di sviluppo e le opzioni per ricaricare le classi che si applicano alle procedure di Marketing Operations, ma ciò non è obbligatorio.

Librerie

IBM Marketing Operations fornisce alcune librerie gratuite e di terze parti; i server delle applicazioni utilizzano anche versioni differenti di queste librerie.

In genere, questo elenco cambia da una release a un'altra. Sono supportate le seguenti librerie di terzi.

- Ant 1.6.5 (ant.jar)
- Axis2 1.5.2 e dipendenze
 - 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)
- API IBM Marketing Operations (affinium_plan.jar)
- API IBM Marketing Platform (unica-common.jar)

Se una procedura o le classi secondarie importate dalla procedura non utilizzano questi package, il loro utilizzo deve essere conforme esattamente ai package forniti da Marketing Operations o dal server delle applicazioni. In questo caso, è necessario rielaborare il codice della procedura se una versione successiva di Marketing Operations aggiorna o non utilizza più una libreria.

Procedure e thread

La procedura deve essere thread-safe per quanto riguarda lo stato. Ciò significa che il suo metodo di esecuzione non può dipendere dalle modifiche di stato interne da una chiamata all'altra. Una procedura non può creare dei thread.

Parametri di configurazione

Quando si installa Marketing Operations Integration Module, il programma di installazione imposta tre proprietà di configurazione. È possibile modificare le proprietà di configurazione per personalizzare il comportamento di Integration Module.

Le proprietà di configurazione per Integration Module sono in **Marketing Operations | umoConfiguration | integrationServices**.

- La proprietà di configurazione **enableIntegrationServices** attiva e disattiva il modulo dei servizi di integrazione.
- Il parametro **integrationProcedureDefinitionPath** contiene il percorso file completo del file XML di configurazione della procedura personalizzata.
Il valore personalizzato è `<IBM_IMS_Home><MarketingOperations_Home>/devkits/integration/examples/src/procedure/procedure-plugins.xml/`.
- Il parametro **integrationProcedureClasspathURL** contiene l'URL del percorso classe delle procedure personalizzate.
Il valore predefinito è `file:/// <IBM_IMS_Home><MarketingOperations_Home>/devkits/integration/examples/classes/`.

Nota: il segno '/' alla fine del percorso `integrationProcedureClasspathURL` è richiesto per caricare correttamente le classi della procedura.

Progettazione

La classe di implementazione della procedura utilizza l'API IBM Marketing Operations per leggere e aggiornare i componenti di Marketing Operations, per aggiornare i servizi e così via. Altri package Java possono essere utilizzati per eseguire altre attività.

Nella progettazione produrre una singola unità di lavoro che funzioni in modo automatico. Idealmente, una procedura esegue delle serie di attività che possono essere pianificate in modo asincrono per essere eseguite successivamente. Questo modello di integrazione detto "fire and forget" genera come risultato il minor carico possibile su entrambi i sistemi.

Nota: solo i metodi e le classi documentate verranno supportate nelle release future di Marketing Operations. Tutti gli altri metodi e tutte le altre classi in Marketing Operations devono considerarsi obsoleti.

Dopo aver codificato e compilato le classi di implementazione della procedura, renderle disponibili a Marketing Operations. Gli script di creazione forniti con i servizi di integrazione di Marketing Operations inseriscono le procedure compilate nell'ubicazione predefinita. Lo step di sviluppo finale consiste nell'aggiornare il file di definizione del plugin della procedura personalizzata utilizzato da Marketing Operations per rilevare le procedure personalizzate.

La procedura deve implementare l'interfaccia **com.unica.publicapi.plan.plugin.procedure.IProcedure** e deve avere un costruttore senza parametri (in genere, il modello JavaBeans). La codifica e l'implementazione di ciascuna procedura vengono eseguite in un IDE Java scelto dal cliente, ad esempio Eclipse, Borland JBuilder o Idea. Il codice di esempio viene fornito con IBM Marketing Operations come SDK nella seguente ubicazione:

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

Ciclo di vita della procedura

Ogni procedura viene eseguita in un ciclo di vita completo.

Il ciclo di vita di runtime di una procedura include i seguenti step.

1. Rilevamento e inizializzazione
2. Selezione (facoltativo)
3. Esecuzione
4. Eliminazione

Rilevamento e inizializzazione

IBM Marketing Operations deve essere a conoscenza di tutte le procedure standard e personalizzate disponibili per una determinata istanza dell'installazione. Questo processo è denominato rilevamento.

Nota: le procedure standard (procedure definite dal team ingegneristico di Marketing Operations) sono note in modo implicito, pertanto non è richiesta alcuna azione per rilevarle.

Le procedure personalizzate sono definite nel file di definizione del plugin della procedura. Il gestore del plugin di Marketing Operations legge questo file durante l'inizializzazione. Per ogni procedura trovata, il gestore del plugin completa i seguenti step.

1. Crea l'istanza della procedura; trasforma il suo stato in INSTANTIATED.
2. Crea un record di controllo della procedura.
3. Se è stata creata l'istanza della procedura, il suo metodo **initialize()** viene chiamato con i parametri di inizializzazione trovati nel relativo file di descrizione del plugin. Se questo metodo genera un'eccezione, lo stato viene registrato e la procedura viene interrotta. Altrimenti, lo stato della procedura diventa INITIALIZED. Ora è pronta per l'esecuzione.
4. Crea un record di controllo della procedura.
5. Se la procedura è stata inizializzata, il suo metodo **getKey()** viene chiamato per determinare la chiave utilizzata dai client per fare riferimento alla procedura. Questa chiave è associata all'istanza e viene salvata per la ricerca successiva.

Selezione

Di tanto in tanto IBM Marketing Operations potrebbe presentare agli utenti un elenco di procedure disponibili, ad esempio, per consentire agli amministratori di configurare un trigger. Marketing Operations presenta questo elenco solo dopo l'inizializzazione della procedura, utilizzando i metodi **getDisplayName()** e **getDescription()** della procedura.

Esecuzione

Dopo l'inizializzazione della procedura, IBM Marketing Operations riceve una richiesta di esecuzione della procedura. Questa richiesta potrebbe presentarsi simultaneamente ad altre procedure (o alla stessa procedura) in esecuzione su altri thread.

Durante il runtime il gestore dell'esecuzione della procedura completa i seguenti step.

1. Avvia una transazione del database.
2. Imposta lo stato della procedura su EXECUTING.
3. Crea un record di controllo della procedura.
4. Richiama il metodo **execute()** della procedura con un contesto di esecuzione e i parametri di esecuzione forniti dal client. L'implementazione del metodo utilizza l'API Marketing Operations come necessario, acquisendo i blocchi di modifica e diffondendo il contesto di esecuzione. Se il metodo run genera un'eccezione, il gestore dell'esecuzione contrassegna la transazione per il rollback.
5. Esegue il commit o il rollback della transazione in base ai risultati dell'esecuzione; imposta lo stato della procedura su EXECUTED.
6. Rilascia i blocchi di modifica in sospeso.
7. Crea un record di controllo della procedura.

Nota: il metodo **execute()** non ha lo scopo di modificare i dati dell'istanza della procedura.

Eliminazione

Quando viene eseguito lo shut down di IBM Marketing Operations, il gestore del plugin della procedura passa in rassegna tutte le procedure caricate. Per ogni procedura trovata, completa i seguenti step.

1. Chiama il metodo destroy() della procedura per consentire alla procedura di eseguire la ripulitura prima dell'eliminazione dell'istanza.
2. Modifica lo stato della procedura in FINALIZED (non può essere eseguita).
3. Crea un record di controllo della procedura.
4. Elimina l'istanza della procedura.

Classi Java chiave

L'SDK di integrazione fornito contiene una serie di Javadoc per l'API pubblica IBM Marketing Operations e le classi di supporto.

Le classi Java più importanti sono elencate di seguito.

- **IProcedure** (com.unica.publicapi.plan.plugin.procedure.IProcedure): interfaccia che tutte le procedure devono implementare. Le procedure attraversano un ciclo di vita ben definito e accedono all'API Marketing Operations per eseguire il lavoro.
- **ITriggerProcedure** (com.unica.publicapi.plan.plugin.procedure.ITriggerProcedure): interfaccia che tutte le procedure trigger devono implementare (interfaccia indicatore).
- **IExecutionContext** (com.unica.publicapi.plan.plugin.procedure.IExecutionContext): interfaccia dell'oggetto di contesto opaco che viene fornita alla procedura dal gestore dell'esecuzione. Questo oggetto dispone di metodi pubblici per registrare e modificare la gestione del blocco. Inoltre, la procedura passa questo oggetto a tutte le chiamate PlanAPI.
- **IPlanAPI** (com.unica.publicapi.plan.api.IPlanAPI): interfaccia dell'API Marketing Operations. Il contesto di esecuzione fornisce un metodo **getPlanAPI()** per richiamare l'implementazione appropriata.

Blocco dei dati

IBM Marketing Operations utilizza uno schema di blocco delle modifiche pessimistico; ciò significa che a un solo utente per volta è consentito l'accesso alle istanze del componente per l'aggiornamento. Per l'utente della GUI, questo blocco viene eseguito a livello di scheda visiva. In alcuni casi, i dati vengono bloccati per un sottoinsieme di un'istanza, ad esempio, una scheda di riepilogo del progetto. In altri casi, i dati vengono bloccati in molte istanze, ad esempio, nella scheda del flusso di lavoro. Dopo che un utente ha acquisito un blocco, tutti gli altri utenti sono limitati all'accesso in sola lettura ai dati correlati.

Per verificare che le modifiche apportate da una procedura a un'istanza del componente o a un gruppo di istanze non vengano sovrascritte per errore da un altro utente, una procedura deve acquisire i blocchi appropriati prima che aggiorni i dati del componente. L'oggetto del contesto di esecuzione passato al metodo **execute()** della procedura viene utilizzato per eseguire il blocco dei dati.

Prima di aggiornare i dati, la procedura deve chiamare il metodo **acquireLock()** del contesto per ciascun blocco richiesto. Ad esempio, se una procedura sta per aggiornare un progetto e il flusso di lavoro associato, deve acquisire il blocco per entrambi.

Se un altro utente ha già il blocco, il metodo **acquireLock()** genera immediatamente un'eccezione **LockInUseException**. Per ridurre al minimo i conflitti, la procedura deve rilasciare il blocco non appena aggiorna l'oggetto.

Il gestore dell'esecuzione rilascia automaticamente i blocchi in sospeso quando viene restituito il metodo **execute**. In ogni caso, i blocchi vengono conservati solo per la durata della transazione del database. Ciò significa che i blocchi scadono se viene superato il timeout di transazione specifico del database.

Nota: i blocchi di modifica non sono uguali alle transazioni del database.

Transazioni della procedura

Il gestore dell'esecuzione della procedura esegue automaticamente il wrapping dell'esecuzione della procedura con una transazione del database, eseguendone il commit o il rollback in base al risultato dell'esecuzione della procedura.

Il wrapping dell'esecuzione della procedura e della transazione del database garantisce che gli aggiornamenti al database IBM Marketing Operations non siano visibili ad altri utenti finché non viene eseguito il commit. Inoltre, rende atomici gli aggiornamenti.

Colui che scrive la procedura deve comunque acquisire i blocchi di modifica necessari per garantire che altri utenti non possano scrivere le modifiche al database prima del completamento dell'esecuzione della procedura.

Comunicazione della procedura

Il metodo **execute()** di una procedura restituisce un codice di stato intero alla tabella di controllo della procedura IBM Marketing Operations. Il metodo **execute()** di una procedura può anche restituire zero o più messaggi alla tabella di controllo della procedura, che vengono registrati e sono persistenti.

Il client potrebbe anche comunicare le informazioni di stato in altro modo.

Registrazione della procedura

IBM Marketing Operations ha un file di log separato per le procedure:
<MarketingOperations_Home>\logs\system.log

Il gestore dell'esecuzione delle procedure registra il ciclo di vita di ciascuna procedura e crea i record di controllo.

- **logInfo()**: scrive un messaggio informativo nel log della procedura.
- **logWarning()**: scrive un messaggio di avviso nel log della procedura.
- **logError()**: scrive un messaggio di errore nel log della procedura.
- **logException()**: esegue il dump della traccia di stack per l'eccezione nel log della procedura.

File di definizione del plugin della procedura

Il file di definizione del plugin della procedura definisce la classe di implementazione, i metadati e altre informazioni sulle procedure personalizzate che devono essere ospitate in IBM Marketing Operations.

Per impostazione predefinita, si presume che la definizione del plugin della procedura sia nel seguente percorso:

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

Questo file è un documento XML contenente le seguenti informazioni.

Procedure: un elenco di zero o più elementi **Procedura**.

Procedura: un elemento che definisce una procedura. Ogni procedura contiene i seguenti elementi.

- **key** (facoltativo): stringa che definisce la chiave di ricerca per la procedura. Questa chiave deve essere unica in tutte le procedure standard (fornite da IBM) e personalizzate, che sono ospitate da una determinata istanza di Marketing Operations. Se non è definita, per impostazione predefinita, viene utilizzata la versione completa dell'elemento **className**. I nomi che iniziano con la stringa "uap" sono riservati per l'utilizzo da parte di IBM Marketing Operations.
- **className** (obbligatorio): nome del package completo della classe della procedura. Questa classe deve implementare la classe `IProcedure` (`com.unica.public.plan.plugin.procedure.IProcedure`).
- **initParameters** (facoltativo): un elenco di zero o più elementi `initParameter`.
initParameter(facoltativo): parametro da passare al metodo `initialize()` della procedura. Questo elemento include il nome, il tipo e gli elementi valore del parametro nidificato.
 - **name**: stringa che definisce il nome del parametro
 - **type**: nome classe facoltativo della classe wrapper Java, che definisce il tipo di valore del parametro. Deve essere uno dei seguenti tipi:
 - `java.lang.String` (valore predefinito)
 - `java.lang.Integer`
 - `java.lang.Double`
 - `java.lang.Calendar`
 - `java.lang.Boolean`

- value: formato stringa del valore dell'attributo in base al relativo tipo

Capitolo 4. API SOAP IBM Marketing Operations

L'API SOAP IBM Marketing Operations offre una vista client di un'istanza di Marketing Operations in esecuzione.

Solo un sottoinsieme di funzioni di Marketing Operations viene mostrato agli utenti. L'API viene utilizzata simultaneamente dagli utenti del web di Marketing Operations, dai trigger e dalle richieste SOAP del servizio web dei servizi di integrazione di Marketing Operations. L'API supporta i seguenti tipi di operazioni.

- Creazione ed eliminazione del componente
- Rilevamento (per tipo di componente, valore attributo e altri valori)
- Analisi del componente (mediante i suoi attributi, link specializzati e altri valori)
- Modifica del componente

Nota: le API Marketing Operations devono essere utilizzate solo dall'amministratore.

Contenuto dell'API SOAP IBM Marketing Operations

Il package `com.unica.publicapi.plan.api` viene fornito con l'API SOAP IBM Marketing Operations.

Questo package offre le interfacce e le eccezioni, inoltre contiene i seguenti tipi di classi.

- Tipi di dati elencati.
- Handle per identificare le istanze di componenti e oggetti.
- Una mappa Java, `AttributeMap`.

La documentazione completa dell'API, inclusi tutti i metodi e i valori possibili, è disponibile facendo clic su **Guida> Documentazione del prodotto** in un'istanza di Marketing Operations ed eseguendo il download del file IBM `<version>PublicAPI.zip`.

Interfacce API SOAP

L'API (Application Programming Interface) SOAP IBM Marketing Operations include **IPlanAPI** e **IExecutionContext**.

L'API SOAP Marketing Operations include le seguenti interfacce.

IPlanAPI

Definisce l'API pubblica per Marketing Operations. Fornisce i metodi per la creazione, il rilevamento e la modifica di oggetti, tra cui cartelle, progetti, programmi, attività del flusso di lavoro e membri del team.

Per i sistemi su cui è abilitata l'integrazione facoltativa con IBM Campaign, fornisce anche i metodi per la creazione, il rilevamento e la modifica delle offerte.

IExecutionContext

Definisce i trigger e i blocchi che eseguono i metodi nell'API.

Metodi API

Per informazioni specifiche sui metodi dell'API pubblica, fare riferimento alla classe iPlanAPI nei file della documentazione dell'API JavaDocs.

Questi file sono disponibili eseguendo l'accesso a Marketing Operations e selezionando **Guida > Documentazione prodotto** da qualsiasi pagina e scaricando il file `<version>PublicAPI.zip`.

Eccezioni comuni API SOAP

Le eccezioni comuni generate dall'API SOAP includono NotFoundException, AuthorizationException, DataException, InvalidExecutionContextException e NotLockedException.

Nel seguente elenco viene indicato perché potrebbero verificarsi queste eccezioni comuni.

- `<object type>NotFoundException`: il sistema non è in grado di restituire l'elemento o l'oggetto specificato.
- `AuthorizationException`: l'utente associato al contesto di esecuzione non è autorizzato per l'operazione richiesta. Questa eccezione può essere generata da qualsiasi metodo API, quindi non è dichiarata.
- `DataException`: si è verificata un'eccezione nel livello del database sottostante in IBM Marketing Operations. Controllare il log SQL per i dettagli.
- `InvalidExecutionContextException`: si è verificato un problema con il contesto di esecuzione passato a un metodo API (ad esempio, il metodo non è stato inizializzato correttamente). Questa eccezione può essere generata da qualsiasi API, quindi non è dichiarata.
- `NotLockedException`: si è tentato di aggiornare i dati del componente senza prima acquisire il blocco richiesto. Vedere il metodo `acquireLock()` dell'interfaccia `IExecutionContext`.

Handle API SOAP

Un handle è un oggetto URL speciale che fa riferimento a una particolare istanza dell'oggetto in un'istanza di IBM Marketing Operations. Gli handle includono il tipo di componente, l'identificativo interno dei dati e un URL di base dell'istanza.

Gli handle utilizzati o generati dall'API possono essere mostrati a un URL completo. È possibile utilizzare l'URL risultante in modi differenti. È possibile utilizzare l'URL per aprire una vista del componente nella GUI di Marketing Operations, inviarlo nei messaggi email o utilizzarlo in un'altra procedura come un parametro.

Gli handle sono validi solo per una particolare istanza del cluster o del servizio di Marketing Operations, ma sono validi per la durata del servizio distribuito. Come risultato, gli handle possono essere salvati in un file per riferimento futuro, ma possono essere utilizzati per accedere ai componenti su un'altra istanza di Marketing Operations. Questa restrizione si applica anche alle istanze sullo stesso server host fisico. Tuttavia, Marketing Operations non fornisce un meccanismo per associare URL di base differenti all'istanza corrente per ricollocare un'istanza su un altro server (ad esempio, se l'apparecchiatura non funziona correttamente).

Gli handle dipendono dal client. Ad esempio, un trigger può passare un handle a una procedura, che lo utilizza come un parametro in una chiamata SOAP a un

sistema di terze parti. Il sistema di terze parti può quindi generare una richiesta SOAP affinché Marketing Operations avvii una procedura per aggiornare un attributo.

I membri della classe dell'handle hanno metodi factory per creare gli handle da vari tipi di URL. Di seguito vengono forniti degli esempi.

Approvazione

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

Asset

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

Cartella degli asset

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

Libreria degli asset

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

Allegato

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

Account finanziario

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

Cartella

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

Fattura

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

Voce della fattura

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

Oggetto di marketing

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

Griglia dell'oggetto di marketing

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

Riga della griglia dell'oggetto di marketing

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

Team del piano

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

Utente del piano

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

Programma

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

Griglia del programma

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

Riga della griglia del programma

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

Progetto

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

Griglia del progetto

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

Riga della griglia del progetto

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

Voce del progetto

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

Fase del flusso di lavoro

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

Attività del flusso di lavoro

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

AttributeMap API SOAP

La classe `AttributeMap` è una mappa Java che contiene solo attributi. L'attributo `<Name>` è la chiave di immissione della mappa, mentre l'array di attributi `<values>` (si osservi il plurale) è il valore di immissione della mappa.

La classe `AttributeMap` include i seguenti campi.

- `<Name>`: il nome programmatico dell'attributo. Questo nome funge da chiave unica per l'accesso all'attributo all'interno dell'istanza del componente in cui si trova.

Nota: `<Name>` non è necessariamente il nome di visualizzazione presentato all'utente nella GUI. Per i componenti creati dai modelli (ad esempio, progetti o attività del flusso di lavoro), il nome dell'attributo è specificato dalla definizione dell'elemento del modello. Il nome dell'attributo deve essere univoco. Per gli altri componenti, il nome dell'attributo in genere è derivato in modo programmatico dall'istanza del componente lato server (ad esempio, mediante l'introspezione Java).

Nota: per convenzione, gli attributi personalizzati includono il nome del modulo in cui è definita la versione modificabile: `<form_name>.<attribute_name>`.

- `Values`: un array di oggetti Java contenente zero o più valori di attributo. Il tipo di ciascun valore deve essere identico e soddisfare il tipo di attributo definito in Marketing Operations. Sono supportati solo i seguenti tipi Marketing Operations e wrapper Java:
 - `AssetLibraryStateEnum`: un valore tipo elencato `AssetLibraryStateEnum`.
 - `AssetStateEnum`: un valore tipo elencato `AssetStateEnum`.
 - `AttachmentTypeEnum`: un valore tipo elencato `AttachmentTypeEnum`.
 - `AttributeMap`: una mappa contenente gli attributi.
 - `BudgetPeriodEnum`: un valore tipo elencato `BudgetPeriodEnum`.
 - `BudgetTypeEnum`: un valore tipo elencato `BudgetTypeEnum`.

- Handle: un riferimento all'istanza di un componente, alla riga della griglia, all'attributo e così via.
- InvoiceStateEnum: un valore tipo elencato InvoiceStateEnum.
- java.io.File: rappresentazione di un file.
- java.lang.Boolean: un valore booleano, True o False
- java.lang.Double: un valore numero decimale a precisione doppia.
- java.lang.Float: un valore numero decimale a precisione singola
- java.lang.Integer: un valore numero intero a 32 bi
- java.lang.Long: un valore numero intero a 64 bit
- java.lang.Object: oggetto Java generico
- java.lang.String: una stringa di zero o più caratteri Unicode
- java.math.BigDecimal: valore numero decimale con segno di precisione arbitraria. Adatto per la valuta; l'interpretazione del valore dipende dalla locale della valuta del client.
- java.math.BigInteger: valore numero intero di precisione arbitraria.
- java.net.URL: un oggetto URL (Universal Resource Locator).
- java.util.ArrayList: elenco di oggetti.
- java.util.Calendar: un valore data-ora per una particolare locale.
- java.util.Date: un valore data-ora. Questo tipo è obsoleto. Utilizzare java.util.Calendar o java.util.GregorianCalendar.

Nota: per implementare la data, gli utenti possono selezionare java.util.Calendar o java.util.GregorianCalendar.

- java.util.GregorianCalendar: GregorianCalendar è una classe secondaria concreta di java.util.Calendar e fornisce il sistema del calendario standard utilizzato nella maggior parte del mondo.
- MonthEnum: un valore tipo elencato MonthEnum.
- ProjectStateEnum: un valore tipo elencato ProjectStateEnum.
- QuarterEnum: un valore tipo elencato QuarterEnum.
- TaskStateEnum: un valore tipo elencato TaskStateEnum.
- WeekEnum: un valore tipo elencato WeekEnum.

I metadati di un attributo (ad esempio, il nome di visualizzazione tradotto e la descrizione) sono definiti dal modello associato all'attributo e alla relativa istanza dell'oggetto padre. Gli attributi forniscono un meccanismo semplice, ma completo per mostrare gli attributi dell'istanza dell'oggetto obbligatori e facoltativi, ad esempio, nome del progetto, codice e data di inizio.

Tipi di dati elencati API SOAP

La API SOAP IBM Marketing Operations supporta i seguenti tipi di dati e valori elencati.

ApprovalMethodEnum

ApprovalMethodEnum definisce i metodi di approvazione validi. Valori possibili:

- SEQUENTIAL
- SIMULTANEOUS

ApprovalStateEnum

ApprovalStateEnum definisce gli stati di approvazione validi. Valori possibili:

- CANCELLED
- COMPLETED
- IN_PROGRESS
- NOT_STATED
- ON_HOLD

AssetLibraryStateEnum

AssetLibraryStateEnum definisce gli stati della libreria degli asset validi. Valori possibili:

- DISABLED
- ENABLED

AssetStateEnum

AssetStateEnum definisce gli stati dell'asset validi. Valori possibili:

- ARCHIVE
- DRAFT
- FINALIZE
- LOCK

AttachmentTypeEnum

AttachmentTypeEnum definisce i tipi di allegato validi. Valori possibili:

- ASSET
- FILE
- URL

BudgetPeriodEnum

BudgetPeriodEnum definisce i periodi di budget possibili. Valori possibili:

- ALL
- MONTHLY
- QUARTERLY
- WEEKLY
- YEARLY

BudgetTypeEnum

BudgetTypeEnum definisce i tipi di budget validi. Valori possibili:

- EFFETTIVO
- ALLOCATED
- COMMITTED
- FORECAST
- TOTAL

ComponentTypeEnum

ComponentTypeEnum identifica i tipi di componenti di Marketing Operations accessibili. Valori possibili:

- 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 definisce gli stati della fattura validi. Valori possibili:

- CANCELLED
- DRAFT
- PAID
- PAYABLE

MonthEnum

MonthEnum definisce i valori validi per il mese.

OfferStateEnum

OfferStateEnum definisce gli stati dell'offerta validi. Valori possibili:

- STATE_OFFER_DRAFT
- STATE_OFFER_PUBLISHED
- STATE_OFFER_RETIRED

ProjectCopyTypeEnum

ProjectCopyTypeEnum definisce i metodi validi per copiare un progetto. Valori possibili:

- COPY_USING_PROJECT_METRICS
- COPY_USING_TEMPLATES_METRICS

ProjectParticipantLevelEnum

ProjectParticipantLevelEnum identifica i ruoli che gli utenti possono avere in un progetto. Valori possibili:

- OWNER
- PARTICIPANT
- REQUESTER

ProjectStateEnum

ProjectStateEnum definisce gli stati validi di progetto e richiesta. Valori possibili:

- ACCEPTED
- CANCELLED
- COMPLETED
- DRAFT
- IN_PROGRESS
- IN_RECONCILIATION
- LATE: il progetto non è stato avviato per la sua data di inizio pianificata.
- NOT_STARTED
- ON_HOLD

- OVERDUE: il progetto non è stato completato prima della sua data di fine pianificata.
- RETURNED
- SUBMITTED

Per ulteriori informazioni sugli stati di progetti e attività, consultare *IBM Marketing Operations - Guida dell'utente*.

QuarterEnum

QuarterEnum definisce i valori validi per i trimestri: Q1, Q2, Q3 e Q4.

TaskStateEnum

TaskStateEnum definisce gli stati validi dell'attività del flusso di lavoro.

Valori possibili:

- ACTIVE
- DISABLED
- FINISHED
- PENDING
- SKIPPED

WeekEnum

WeekEnum definisce i valori validi per le settimane in un anno, da WEEK_1 a WEEK_53.

Prima di contattare il supporto tecnico di IBM

Se si riscontra un problema che non può essere risolto consultando la documentazione, il contatto designato per l'assistenza della propria azienda può contattare il supporto tecnico di IBM. Utilizzare queste linee guida per garantire una risoluzione efficiente e valida del problema.

Se non si ha il ruolo di contatto designato per l'assistenza per la propria società, contattare l'amministratore IBM per informazioni.

Nota: Il supporto tecnico non scrive o crea script API. Per supporto nell'implementazione delle offerte API, contattare i Servizi professionali IBM.

Informazioni da raccogliere

Prima di contattare il supporto tecnico di IBM, raccogliere le seguenti informazioni:

- Una breve descrizione della natura del problema.
- Messaggi di errore dettagliati visualizzati al verificarsi del problema.
- Step dettagliati per riprodurre il problema.
- File di log, file di sessione, file di configurazione e file di dati correlati.
- Informazioni sull'ambiente di sistema e del prodotto , che è possibile ottenere come descritto in "Informazioni sul sistema".

Informazioni di sistema

Quando si contatta il supporto tecnico di IBM, è possibile che venga richiesto di fornire informazioni relative al proprio ambiente.

Se il problema non impedisce di eseguire l'accesso, la maggior parte di queste informazioni è disponibile nella pagina Informazioni su, che fornisce informazioni sulle applicazioni IBM installate.

È possibile accedere alla pagina Informazioni selezionando **Guida > Informazioni**. Se la pagina Informazioni su non è accessibile, controllare il file `version.txt` che si trova nella directory di installazione dell'applicazione.

Informazioni sul contatto per il supporto tecnico di IBM

Per informazioni su come contattare il supporto tecnico di IBM, visitare il sito web del supporto tecnico del prodotto IBM: (http://www.ibm.com/support/entry/portal/open_service_request).

Nota: Per inserire una richiesta di supporto, è necessario eseguire l'accesso tramite un account IBM. Questo account deve essere collegato al proprio numero cliente IBM. Per ulteriori informazioni relative all'associazione del proprio account al numero cliente IBM, consultare **Risorse del supporto>Supporto software autorizzato** nel portale del supporto.

Informazioni particolari

Queste informazioni sono state sviluppate per prodotti e servizi offerti negli Stati Uniti.

IBM può non offrire i prodotti, i servizi o le funzioni presentati in questo documento in altri paesi. Consultare il proprio rappresentante locale IBM per informazioni sui prodotti ed i servizi attualmente disponibili nella propria zona. Qualsiasi riferimento ad un prodotto, programma o servizio IBM non implica o intende dichiarare che solo quel prodotto, programma o servizio IBM può essere utilizzato. Qualsiasi prodotto funzionalmente equivalente al prodotto, programma o servizio che non violi alcun diritto di proprietà intellettuale IBM può essere utilizzato. È tuttavia responsabilità dell'utente valutare e verificare il funzionamento di eventuali prodotti, programmi o servizi non IBM.

IBM può avere applicazioni di brevetti o brevetti in corso relativi all'argomento descritto in questo documento. La fornitura di tale documentazione non garantisce alcuna licenza su tali brevetti. Chi desiderasse ricevere informazioni relative alle licenze può rivolgersi per iscritto a:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Per richieste di licenze relative ad informazioni double-byte (DBCS), contattare il Dipartimento di Proprietà Intellettuale IBM nel proprio paese o inviare richieste per iscritto a:

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

Il seguente paragrafo non è valido per il Regno Unito o per tutti i paesi le cui leggi nazionali siano in contrasto con le disposizioni in esso contenute:
L'INTERNATIONAL BUSINESS MACHINES CORPORATION FORNISCE QUESTA PUBBLICAZIONE "NELLO STATO IN CUI SI TROVA", SENZA ALCUNA GARANZIA, ESPLICITA O IMPLICITA, IVI INCLUSE EVENTUALI GARANZIE DI COMMERCIALIZZABILITÀ ED IDONEITÀ AD UNO SCOPO PARTICOLARE. Alcuni stati non consentono la rinuncia ad alcune garanzie espresse o implicite in determinate transazioni, pertanto, la presente dichiarazione può non essere applicabile.

Queste informazioni potrebbero includere inesattezze tecniche o errori tipografici. Le modifiche alle presenti informazioni vengono effettuate periodicamente; tali modifiche saranno incorporate nelle nuove pubblicazioni della pubblicazione. IBM può effettuare miglioramenti e/o modifiche ai prodotti e/o ai programmi descritti nella presente pubblicazione in qualsiasi momento senza preavviso.

Qualsiasi riferimento in queste informazioni a siti Web non IBM sono fornite solo per convenienza e non servono in alcun modo da approvazione di tali siti Web. I materiali presenti in tali siti Web non sono parte dei materiali per questo prodotto IBM e l'utilizzo di tali siti Web è a proprio rischio.

IBM può utilizzare o distribuire qualsiasi informazione fornita in qualsiasi modo ritenga appropriato senza incorrere in alcun obbligo verso l'utente.

I licenziatari di questo programma che desiderano avere informazioni allo scopo di abilitare: (i) lo scambio di informazioni tra i programmi creati indipendentemente e gli altri programmi (incluso il presente) e (ii) il reciproco utilizzo di informazioni che sono state scambiate, dovrebbero contattare:

IBM Corporation
B1WA LKG1
550 King Street
Littleton, MA 01460-1250
U.S.A.

Tali informazioni possono essere disponibili, in base ad appropriate clausole e condizioni, includendo in alcuni casi, il pagamento di una tassa.

Il programma concesso in licenza descritto nel presente documento e tutto il materiale concesso in licenza disponibile sono forniti da IBM in base alle clausole dell'Accordo per Clienti IBM (IBM Customer Agreement), dell'IBM IPLA (IBM International Program License Agreement) o qualsiasi altro accordo equivalente tra le parti.

Qualsiasi dato sulle prestazioni qui contenuto è stato determinato in un ambiente controllato. Di conseguenza, i risultati ottenuti in altri ambienti operativi potrebbero variare in modo significativo. Alcune misurazioni possono essere state effettuate su sistemi del livello di sviluppo e non vi è alcuna garanzia che tali misurazioni resteranno invariate sui sistemi generalmente disponibili. Inoltre, alcune misurazioni possono essere state stimate tramite estrapolazione. I risultati effettivi possono quindi variare. Gli utenti del presente documento dovranno verificare i dati applicabili per i propri ambienti specifici.

Le informazioni relative a prodotti non IBM sono ottenute dai fornitori di quei prodotti, dagli annunci pubblicati o da altre fonti disponibili al pubblico. IBM non ha testato quei prodotti e non può confermarne l'accuratezza della prestazione, la compatibilità o qualsiasi altro reclamo relativo ai prodotti non IBM. Le domande sulle capacità dei prodotti non IBM dovranno essere indirizzate ai fornitori di tali prodotti.

Tutte le dichiarazioni relative all'orientamento o alle intenzioni future di IBM sono soggette a modifica o a ritiro senza preavviso e rappresentano unicamente mete ed obiettivi.

Tutti i prezzi IBM mostrati sono i prezzi al dettaglio suggeriti da IBM, sono attuali e soggetti a modifica senza preavviso. I prezzi al fornitore possono variare.

Queste informazioni contengono esempi di dati e report utilizzati in quotidiane operazioni aziendali. Per illustrarle nel modo più completo possibile, gli esempi includono i nomi di individui, società, marchi e prodotti. Tutti questi nomi sono fittizi e qualsiasi somiglianza con nomi ed indirizzi utilizzati da organizzazioni di impresa realmente esistenti è puramente casuale.

LICENZA SUL DIRITTO D'AUTORE:

Queste informazioni contengono programmi applicativi di esempio in linguaggio sorgente, che illustrano tecniche di programmazione su varie piattaforme operative. È possibile copiare, modificare e distribuire questi programmi di esempio sotto qualsiasi forma senza alcun pagamento alla IBM, allo scopo di sviluppare, utilizzare, commercializzare o distribuire i programmi applicativi in conformità alle API (application programming interface) a seconda della piattaforma operativa per cui i programmi di esempio sono stati scritti. Questi esempi non sono stati testati in maniera approfondita e tenendo conto di tutte le possibili condizioni. La IBM, quindi, non può garantire o sottintendere l'affidabilità, l'utilità o il funzionamento di questi programmi. I programmi di esempio vengono forniti "NELLO STATO IN CUI SI TROVANO" e senza alcun tipo di garanzia. IBM declina ogni responsabilità per eventuali danni derivanti dall'uso degli stessi.

Se si visualizzano tali informazioni come softcopy, non potranno apparire le fotografie e le illustrazioni a colori.

Marchi

IBM, il logo IBM e ibm.com sono marchi di International Business Machines Corp., registrati in molte giurisdizioni nel mondo. Altri nomi di prodotti e servizi potrebbero essere marchi di IBM o di altre società. Un elenco attuale di marchi IBM è disponibile sul Web all'indirizzo su Informazioni relative a "Copyright and trademark information" all'indirizzo www.ibm.com/legal/copytrade.shtml.

Normativa sulla privacy e termini di utilizzo - Considerazioni

I prodotti software IBM, incluso il software come le soluzioni di servizi ("Offerta software") possono utilizzare cookie o altre tecnologie per raccogliere informazioni sull'utilizzo del prodotto, per migliorare l'esperienza dell'utente finale, per personalizzare le interazioni con l'utente o per altri scopi. Un cookie è un insieme di dati che un sito web può inviare al browser che viene memorizzato nel computer come un tag che lo identifica. In molti casi questi cookie non raccolgono informazioni personali. Se un'offerta software consente di raccogliere informazioni personali mediante i cookie e tecnologie simili, saranno fornite le seguenti specifiche.

In base alle configurazioni distribuite questa offerta software può utilizzare cookie di sessione e cookie persistenti che identificano il nome utente di ciascun utente e altre informazioni personali per la gestione della sessione, per fornire una migliorata usabilità all'utente o per tenere traccia dell'utilizzo o per scopi funzionali. Questi cookie possono essere disabilitati, ma disabilitando si eliminerà anche la funzione che essi abilitano.

Varie autorità regolano la raccolta di informazioni personali attraverso cookie e tecnologie simili. Se le configurazioni distribuite per questa offerta software forniscono la possibilità di raccogliere informazioni personali degli utenti finali mediante cookie e altre tecnologie, è necessario richiedere un parere legale relativo alle leggi in materia di tale raccolta di dati compreso qualsiasi requisito per ottenere il consenso quando necessario.

IBM richiede di (1) fornire un chiaro e visibile collegamento alle condizioni di utilizzo del sito web del cliente (ad esempio, normative sulla privacy) compreso un collegamento alla raccolta dei dati e alle pratiche d'uso dei clienti e di IBM, (2) notificare che i cookie e i beacon gif/web trasparenti siano collocati sul computer

del visitatore da IBM per conto del cliente unitamente ad una descrizione dello scopo di tali tecnologia e (3) per estensione richiesta dalla legge, ottenere il consenso dai visitatori del sito web prima del posizionamento di cookie e beacon gif/web trasparenti collocati dal cliente o da IBM per conto del cliente sui dispositivi dei visitatori del sito web.

Per informazioni sull'utilizzo di varie tecnologie compreso i cookie, a tale scopo consultare la normativa sulla privacy online IBM all'indirizzo:
<http://www.ibm.com/privacy/details/us/en> e fare riferimento alla sezione "Cookies, Web Beacons and Other Technologies."



Stampato in Italia