

SAP BusinessObjects Business Intelligence プラットフォーム
文書バージョン: 4.2 Support Package 2 – 2016-03-07

SAP Crystal Reports RESTful Web サービス開発者ガイド



目次

1	ドキュメント履歴.....	4
2	はじめに.....	5
2.1	SAP Crystal Reports RESTful Web サービスの概要.....	5
3	RESTful Web サービスの基本.....	6
3.1	RESTful Web サービス要求を行う.....	6
	要求ヘッダの作成.....	10
	要求本文の作成.....	12
	応答ヘッダーの解釈.....	12
	応答本文から XML 形式への変換.....	14
	応答本文から JSON 形式への変換.....	19
	XML と JSON の属性の比較.....	24
	多言語データの操作.....	27
3.2	RESTful Web サービス要求の取得.....	28
	CMC によるベース URL の取得.....	28
	プログラムによるベース URL の取得.....	28
3.3	認証.....	29
	ユーザー名とパスワードからログオントークンを取得する.....	30
	シリアライズセッションまたはセッショントークンからログオントークンを取得する.....	32
	Active Directory シングル サインオン (AD SSO) アカウントを使用してログオントークンを取得する.....	34
	信頼できる認証を使用して、ログオントークンを取得する.....	35
	XML エンコードテキストからログオントークンへの変換.....	37
	要求ヘッダーにログオントークンを追加する.....	38
	HTTP 基本認証の使用.....	39
	BI platform からログオフする.....	40
	他の SDK から取得した認証済みセッションの使用.....	41
3.4	レポート ID の取得.....	43
	プログラムからレポート ID を取得する.....	43
4	SAP Crystal Reports RESTful Web サービス API の使用.....	45
4.1	共通 URL.....	45
4.2	レポートの URL.....	46
4.3	レポートメタデータ.....	47
4.4	レポートインスタンス.....	54
	既存のレポートから一時レポートインスタンスを作成する.....	56

4.5	エクスポート	57
	文字区切り値 (CSV) パラメータ	59
	Microsoft Excel パラメータ	60
	Microsoft Excel データのみパラメータ	62
	リッチテキスト形式 (RTF) パラメータ	63
	タブ区切りテキスト (TTX) パラメータ	64
	XML パラメータ	65
4.6	インタラクティブパラメータ	65
4.7	OData プロトコル	67
	OData サービスドキュメントへのアクセス	67
	総計	68
	グループ	70
	メタデータ	73
	行	75

1 ドキュメント履歴

下の表は、最も重要なドキュメント変更の概要です。

表 1:

バージョン	日付	変更
SAP BusinessObjects Business Intelligence プラッ トフォーム 4.2	2015 年 11 月	節「SAP Crystal Reports RESTful Web サービス API の使用」に節「レポートメタデー タ」が追加されました。
SAP BusinessObjects Business Intelligence プラッ トフォーム 4.2 SP2	2016 年 3 月	注を更新: 要求ヘッダの作成 [10 ページ]

2 はじめに

このオンラインヘルプは、SAP Crystal Reports RESTful Web サービスを使用して独自の Web アプリケーションまたはデスクトップアプリケーションを開発する際に役立つように作成されています。このガイドには、SAP Crystal Reports RESTful Web サービスの実装に使用できる例やプログラミングリファレンスが含まれます。

2.1 SAP Crystal Reports RESTful Web サービスの概要

SAP Crystal Reports RESTful Web サービスを使用すると、SAP BusinessObjects Business Intelligence プラットフォームリポジトリ内で管理されている SAP Crystal Reports for Enterprise レポートデータをモバイルデバイスや Web 対応テクノロジーで利用したり、それらに埋め込むことができます。XML 形式または JSON 形式でレポートコンテンツをフェッチしたり、RESTful API や OData サービスを使用してレポートを操作することができます。RESTful Web サービスでは、任意の開発言語を使用してアプリケーションを作成できます。

SAP Crystal Reports RESTful Web サービスを使用すると、以下の機能を実現できます。

- さまざまなファイルタイプにレポートをエクスポートする
- レポートのメタデータを取得する
- データ行や計算行を取得する
- データ行をレポートにプッシュする
- レポートのパラメータを取得および設定する
- XML 形式または JSON 形式でデータを取得する

i 注記

RESTful Web サービスを使用する場合、SAP Crystal Reports for Enterprise で作成されたレポートのみを操作できます。

3 RESTful Web サービスの基本

このセクションでは、RESTful Web サービス API の基本について説明します。これらの基本は、SAP Business Intelligence platform RESTful Web サービス API から提供される機能と重複します。

詳細については、*Business Intelligence Platform RESTful Web Services Developer Guide* を参照してください。

3.1 RESTful Web サービス要求を行う

Business Intelligence プラットフォーム RESTful Web サービス SDK にアクセスするには、RESTful Web サービスをホストする URL に HTTP 要求を送信します。RESTful Web サービスは要求を処理し、要求された情報を含む応答を返します。HTTP 要求をサポートする任意のプログラミング言語またはツールを使用して、RESTful Web サービスにアクセスできます。RESTful Web サービスは、HTTP 標準と AtomPub 仕様に準拠しますが、カスタム属性も含まれます。

要求は、要求ヘッダーと要求本文の 2 つのメインコンポーネントで構成されます。要求ヘッダーは、要求本文の形式、受け入れられる応答形式のほか、優先言語やログオントークンなどのカスタム設定を定義します。要求本文は空白のままか、要求を行うために必要な追加情報が含まれていることもあります。たとえば、認証要求は、ユーザー名とパスワードを XML 形式で要求本文に渡します。

RESTful Web サービス要求を行うには、以下の情報が必要です。

- URL - RESTful Web サービスをホストする URL。
- メソッド - GET、PUT、POST、DELETE など、要求の送信に使用する HTTP メソッドのタイプ。
- 要求ヘッダー - 要求を記述する属性。
- 要求本文 - 要求の処理に使用される追加情報。

要求が処理されると、応答を受け取ります。応答には、要求した情報のほか、次のステップに必要な補助情報が含まれます。たとえば、応答には、以降の要求の要求本文にデータを設定するための XML テンプレートや、親フォルダ、子フォルダ、追加情報のページ、関連リンクなどの関連する RESTful URL へのリンクが含まれることがあります。RESTful 応答から提供される情報に従って、要求したデータに移動したり、以降の要求に必要なテンプレートを取得することができます。

Business Intelligence プラットフォームの RESTful Web サービス応答は、BI platform クライアント アプリケーションの機能によって XML または JSON の形式にすることができます。

RESTful Web サービスの応答には、以下の 2 つのメインコンポーネントが含まれます。

- 応答ヘッダー - 応答の形式を記述する属性のリスト。HTTP 応答コードを含みます。
- 応答本文 - 要求した情報、および後続の要求を完了するための追加情報。

このドキュメントの例では、各 RESTful 要求に必要な URL、メソッド、要求ヘッダー属性、および要求本文を定義します。HTTP 要求をサポートする任意のプログラミング言語またはツールを使用して、RESTful Web サービスにアクセスできます。



例

/login/long API を使用する RESTful POST 要求と XML 形式を使用する応答

この例では、BI platform リポジトリにログオンする RESTful 要求を示します。

要求

URL: `http://localhost:6405/biprws/logon/long`

メソッド: POST

要求ヘッダー属性:

属性	値
Content-Type	application/xml
Accept	application/xml

要求本文:

```
<attrs xmlns="http://www.sap.com/rws/bip">
  <attr name="userName" type="string">username</attr>
  <attr name="password" type="string">password</attr>
  <attr name="auth" type="string"
possibilities="secEnterprise, secLDAP, secWinAD">secEnterprise</attr>
</attrs>
```

応答

応答ヘッダー:

属性	値
Status code	200 OK
Server	Apache-Coyote/1.1
X-SAP-LogonToken	"COMMANDCOM-LCM: 6400@{3&2=5542,U3&p=40680.8979564815,Y7&4F=12,U 3&63=secEnterprise, 0P&66=60,03&68=secEnterprise:Administrator, 0P&qe=100,U3&vz=TZnoIE2yQyeLCkAlnHtaaYUHon5.p0y TkSaUiLC8SSM,UP}"
Date	Tue, 17 May 2011 21:33:03 GMT
Content-Type	application/xml
Content-Length	586

応答本文:

```
<entry xmlns="http://www.w3.org/2005/Atom">
  <author><name>@COMMANDCOM-LCM: 6400</name></author>
  <id>tag:sap.com,2010:bip-rs/logon/long</id>
  <title type="text">Logon Result</title>
  <updated>2011-05-17T21:33:03.471Z</updated>
  <content type="application/xml">
    <attrs xmlns="http://www.sap.com/rws/bip">
      <attr name="logonToken" type="string">COMMANDCOM-LCM: 6400@{3&am  
2=5542,U3&am...YUHon5.p0yTkSaUiLC8SSM,UP}</attr>
    </attrs>
  </content>
</entry>
```

例

/logon/long API を使用する RESTful GET および POST 要求と JSON 形式を使用する応答

この例では、名前およびパスワードと認証タイプを入力するために使用する JSON 形式の要求本文を GET 要求を使用して取得してから、POST 要求を使用して BI platform リポジトリからログイントークンを取得する RESTful 要求を示します。

要求

URL: `http://localhost:6405/biprws/logon/long`

メソッド: GET

要求ヘッダー属性:

属性	値
Accept	application/json

GET 要求の後に JSON 形式で返される要求本文は、次のとおりです。

```
{"userName":"","password":"","auth":"secEnterprise"}
```

要求本文は、POST 要求として送信される前に、次のコード スニペットに示すように名前ラベル (例: BOEuser) とパスワード (例: BOEPassword999) が含まれます。

```
{"userName":"BOEuser","password":"BOEPassword999","auth":"secEnterprise"}
```

i 注記

auth のデフォルト値は secEnterprise です。使用できる認証タイプは、次のとおりです。

- secEnterprise - Enterprise 認証
- secLDAP - Lightweight Directory Access Protocol 認証
- secWinAD - Windows Active Directory 認証
- secSAPR3 - SAP 認証

POST 要求後の応答ヘッダー:

属性	値
Status code	200 OK
Server	Apache-Coyote/1.1
X-SAP-LogonToken	:"COMMANDCOM-LCM: 6400@{3&2=5571,U3&p=40897.0049317824,Y7&4F=12,U 3&63=secEnterprise, OP&66=60,03&68=secEnterprise:BOEuser, OP&qe=100,U3&vz=odiw9uLc1kVIJf9lggLFEPAX3qsFWB T1LkdE2DTGhY,UP}"
日付	Tue, 17 December 2011 21:33:03 GMT
Content-Type	application/json
Content-Length	204

JSON 形式の応答本文:

```
{ "logonToken": "COMMANDCOM-LCM:
6400@{3&2=5571,U3&p=40897.0049317824,Y7&4F=12,U3&63=secEnterprise,
0P&66=60,03&68=secEnterprise:Administrator,0P&qe=100,U3&vz=odiw9uLc1kVlJf
9lggLFEPAX3qsFWBT1Lkde2DTGhY,UP}" }
```

例

RESTful infostore JSON 形式の要求

ここでは、RESTful 要求の例として、GET 要求と /infostore API およびログイントークンを使用して、JSON 形式で返される情報を BI platform リポジトリから取得します。

要求

URL: `http://commandcom-lcm:6405/biprws/infostore`

メソッド: GET

要求ヘッダー属性:

属性	値
Accept	application/json
X-SAP-LogonToken	COMMANDCOM-LCM: 6400@{3&2=5542,U3&p=40680.897...UiLC8SS

要求本文: (空白)

応答

応答ヘッダー:

属性	値
Status code	200 OK
Server	Apache-Coyote/1.1
X-SAP-LogonToken	COMMANDCOM-LCM: 6400@{3&2=5542,U3&p=40680.897...UiLC8SSM,UP}
日付	Tue, 17 December 2011 21:33:03 GMT
Content-Type	application/json
Content-Length	6919

JSON 形式に設定される応答本文。次のコード スニペットでは、簡略化のために(/)や(")などのエスケープ文字は削除されています。

```
{
  "_metadata":
  { "uri": "http://localhost:9998/biprws/infostore/Root%20Folder/children?
page=1&page=1&pageSize=3",
    "first":
    { "_deferred":
      { "uri": "http://localhost:9998/biprws/infostore/Root%20Folder/children?
page=1&page=1&pageSize=3"
    }
  },
}
```

```
{
  "next":
    { "__deferred":
      { "uri": "http://localhost:9998/biprws/infostore/Root%20Folder/children?
page=2&pageSize=3"}
    },
  "last":
    { "__deferred":
      { "uri": "http://localhost:9998/biprws/infostore/Root%20Folder/children?
page=3&pageSize=3"}
    },
  "entries":
    [
      { "_metadata":
        { "uri": "http://localhost:9998/biprws/infostore/4005"},
          "id": 4005,
          "cuid": "FnKsrkkctAcA8BAAALB7kkQAADAFzVMX",
          "name": "Data Federation",
          "type": "Folder"
        },
      { "_metadata":
        { "uri": "http://localhost:9998/biprws/infostore/3931"},
          "id": 3931,
          "cuid": "Ac1aKZlztj5VJmMQi5Lda53s",
          "name": "LCM",
          "type": "Folder"
        },
      { "_metadata":
        { "uri": "http://localhost:9998/biprws/infostore/5056"},
          "id": 5056,
          "cuid": "Acu9FvxWBZ9Htt0_08a25b4",
          "description": "",
          "name": "Monitoring Report Sample",
          "type": "Folder"
        }
    ]
}
```

関連情報

RESTful Web サービス要求の取得 [28 ページ]

3.1.1 要求ヘッダの作成

HTTP 要求の要求ヘッダには、要求を記述する一連の属性が格納されます。RESTful Web サービス SDK は、標準 HTTP 属性セットとカスタム属性を認識します。

次の表で、RESTful Web サービスが認識する要求ヘッダについて説明します。

属性	説明
Accept	指定可能なコンテンツの種類。要求本文のコンテンツを指定します。この値を使用して、形式を指定します。

属性	説明
	<p>指定可能な値の例:</p> <ul style="list-style-type: none"> • application/xml • application/atom+xml • application/json
Accept-Charset	指定可能な文字セット。
Accept-Encoding	指定可能な文字エンコード。
Accept-Language	システムメッセージとエラーメッセージの取得に使用される優先言語。これは、BI platform の製品ロケール(PL)に対応します。
Content-Length	要求本文に含まれる文字数。
Content-Type	<p>要求本文のコンテンツの種類。</p> <p>指定可能な値の例:</p> <ul style="list-style-type: none"> • application/xml • application/atom+xml • application/atom+xml;type=feed
Host	RESTful Web サービスのインターネットホストとポート番号。
X-SAP-LogonToken	<p>認証プロセスから受け取ったログオントークンの値。</p> <div> <p>i 注記</p> <p>ログオントークンを要求ヘッダに追加する場合は、引用符で囲む必要があります。例:</p> <pre>X-SAP-LogonToken: "COMMANDCOM-LCM: 6400@{3&2=5328,U3&p=40676 . 8926203819,Y7&4F=12,U3&63=secEnterprise , 0P&66=60,03&68=secEnterprise:Administrator ,0P&qe=100,U3&vz=IVD21LbMCB0eRiI4atz9sNL18Ux5anRBdYB9fFv5NrY,UP}"</pre> </div>
X-SAP-PVL	BI platform コンテンツの取得に使用される優先言語。これは、優先表示言語 (PVL) に対応します。

i 注記

Accept、*Content-Type*、および *X-SAP-LOGONTOKEN* などの属性は、要求ヘッダには必須です。

関連情報

[認証 \[29 ページ\]](#)

3.1.2 要求本文の作成

要求本文には、RESTful Web サービスが要求を完了するために必要な情報が含まれます。たとえば、認証要求の要求本文には、ユーザー名とパスワードを含むログオン情報が含まれます。これは、ログオン要求を受け入れるか拒否するかを判断するために必要な情報を認証 URL に提供します。

要求本文の形式を定義するには、要求ヘッダーで属性を設定します。形式を指定するには、メッセージヘッダーの `Content-Type` 属性を設定します。

3.1.3 応答ヘッダーの解釈

応答ヘッダーには、要求が成功したかどうかを示す属性、および応答本文のコンテンツを示す属性が含まれます。応答ヘッダーの属性の大部分は HTTP 標準に含まれます。ただし、`X-SAP-LogonToken` ヘッダー属性は、BI platform でのみ使用されるカスタム属性です。

ステータスコード

ステータスコードには、要求が成功したかどうかを示す標準 HTTP ステータスコードが含まれます。

HTTP 応答コード	エラー	説明
400	不正な要求	要求されたリソースは存在しますが、要求にエラーがあります。
401	ログオンの失敗または無効なセッション	ログオンに失敗しました。ユーザー名、パスワード、およびサーバー名が正しいことを確認してください。
403	アクセス拒否	要求されたリソースを操作する権限がありません。現在のセッションが期限切れになっている可能性があります。ログオンして新しいセッションを取得してください。
404	サービスがない	要求されたサービスは、RESTful Web サービス SDK から提供されていません。
405	要求メソッドが無効	リソースがサポートしていないメソッドを使用して要求が行われました。たとえば、読み取り専用リソースに対して PUT 要求を行いました。

HTTP 応答コード	エラー	説明
406	受容不可	要求されたリソースは、要求ヘッダーの Accept 属性で指定されているコンテンツタイプを生成できません。
408	BI platform サーバータイムアウト	要求の待機中にサーバーがタイムアウトになりました。
415	サポートされていないメディアタイプ	サーバーまたはリソースがサポートしていないメディアタイプが要求に含まれています。
500	RESTful Web サービス内部エラー	未分類のエラーが発生しました。詳細については、応答本文を参照してください。
503	RESTful Web サービスプラグインが見つからない	RESTful Web サービスを利用できません。RESTful Web サービスが正しく構成されていることを確認してください。

Server

要求の処理に使用されたサーバー。

Date

応答日時。

Content-Type

応答本文の形式。たとえば、ほとんどの Web サービス応答は、応答本文が XML 形式であることを示すために、値 `application/xml` を使用しています。

Content-Length

応答本文の長さ。

Transfer-Encoding

メッセージの転送に使用されたエンコードのタイプ。

Content-Location

リソースの検索に使用できる代替リンク。

X-SAP-LogonToken

BI platform へのアクセスが認証されていることを証明するために、後続の要求で利用できるトークン。認証要求は、応答ヘッダーで X-SAP-LogonToken カスタム属性を返します。ログオントークンは、引用符で囲んで後続の要求の要求ヘッダーに挿入します。

i 注記

認証応答の応答本文には、X-SAP-LogonToken 値のコピーが返されます。ただし、応答本文は XML 形式であり、ログオントークンは XML エンコードバージョンに変換されます。このログオントークンのコピーを使用するには、元の形式に変換する必要があります。

関連情報

[XML エンコードテキストからログオントークンへの変換 \[37 ページ\]](#)

3.1.4 応答本文から XML 形式への変換

Business Intelligence プラットフォーム RESTful Web サービス SDK は、Atom 仕様に従って XML 形式で応答を提供します。この仕様は <http://www.w3.org> から入手できます。このセクションでは、RESTful Web サービスに XML タグがどのように適用されるかを説明します。次の画面は、BI launchpad が一般的な /infostore 要求への応答として XML データをどのように返すかを示しています。

```

<?xml version="1.0" ?>
<feed xmlns="http://www.w3.org/2005/Atom">
  <id>tag:sap.com,2010:bip-rs/infostore</id>
  <title type="text">InfoStore {@COMMANDCOM-LCM:6400}</title>
  <updated>2012-02-02T22:35:11.975Z</updated>
  <link href="http://localhost:6405/biprws/infostore/4/children?page=1&pageSize=5" rel="self"/>
  <link href="http://localhost:6405/biprws/infostore/4/children?page=1&pageSize=5" rel="first"/>
  <link href="http://localhost:6405/biprws/infostore/4/children?page=2&pageSize=5" rel="next"/>
  <link href="http://localhost:6405/biprws/infostore/4/children?page=7&pageSize=5" rel="last"/>
  <entry>
    <title type="text">Alert Notifications</title>
    <id>tag:sap.com,2010:bip-rs/ARZB.BFCQk9PqacDpcFwo1w</id>
    <author>
      <name>System Account</name>
    </author>
    <link href="http://localhost:6405/biprws/infostore/Alert%20Notifications" rel="alternate"/>
    <content type="application/xml">
      <attrs xmlns="http://www.sap.com/rws/bip">
        <attr name="id" type="int32">64</attr>
        <attr name="cuid" type="string">ARZB.BFCQk9PqacDpcFwo1w</attr>
        <attr name="description" type="string">Description here</attr>
        <attr name="name" type="string">Alert Notifications</attr>
        <attr name="type" type="string">Folder</attr>
      </attrs>
    </content>
  </entry>
  <entry>
    <title type="text">Application Folder</title>
    :
  </content>
</entry>
</feed>

```

<feed>

<feed> 要素は、<entry> 要素のリストを定義します。JSON では、中かっこの { と } を使用して応答を囲みます。

```

<feed xmlns="http://www.w3.org/2005/Atom">
  <entry> ... </entry>
  <entry> ... </entry>
  ...
</feed>

```

<entry>

単一の項目。<entry> タグには、xmlns 属性を含めることができます。

```

<entry xmlns="http://www.w3.org/2005/Atom">
  ...
</entry>

```

<author>

アクセスされたリソースの所有者。<author> 要素には、リソースの所有者の名前を定義する <name> 要素が含まれます。次の要素は、リソースの所有者が System Accountであることを示します。

```
<author><name>System Account</name></author>
```

<id>

リソースの一意の ID。

```
<id>tag:sap.com,2010:bip-rs/AdoctK9h1sBHp3I6uG0Sh7M</id>
```

<title>

リソースの名前。この例は、リソースの名前が Application Folderであることを示します。

```
<title type="text">Application Folder</title>
```

<updated>

リソースが最後に更新された日時。

```
<updated>2011-04-14T10:27:50.672Z</updated>
```

<link>

link 要素は、他の RESTful Web サービス要求で利用できる URL へのリンクを定義します。これには、親フォルダ、子フォルダ、要求に関連する追加情報などを入れることができます。これらのリンクをたどることで、BI platform リポジトリを参照できます。

リンクタグの href 属性はハイパーリンクを定義し、rel 属性はリンクのタイプを示します。次のリストでは、rel 属性に指定できる値を説明します。

<link> 関連する属性名	説明
self	この URL へのリンク。
first	結果の最初のページへのリンク。

<link> 関連する属性名	説明
next	結果の次のページへのリンク。
previous	結果の前のページへのリンク。
last	結果の最後のページへのリンク。
alternate	同じリソースへの別のリンク。
up	現在のリソースの親へのリンク。
related	関連リソースへのリンク。
http://www.sap.com/rws/bip#children	現在のリソースの子へのリンク。
http://www.sap.com/rws/bip#opendocument	OpenDocument でリソースを表示するためのリンク。
http://www.sap.com/rws/bip#schedule	リソースをスケジュールするためのリンク。

たとえば、次のリンク要素は、結果の次のページへのリンクを示します。

```
<link href="http://localhost:6405/biprws/infostore/Root%20Folder/children?
page=3&pageSize=3" rel="next"></link>
```

応答は、ドキュメントへのリンクを提供するほかに、OpenDocument を使用してドキュメントを表示するための OpenDocument URL も提供します。

```
<link href="http://localhost:8080/BOE/OpenDocument/opendoc/openDocument.jsp?
sIDType=CUID&iDocID=Aa0U0jQbtKxCn.D3JDL0aHs" rel="http://www.sap.com/rws/
bip#opendocument" title="OpenDocument">
```

OpenDocument の詳細については、*OpenDocument* によるドキュメントの表示を参照してください。

i 注記

この SDK から取得したログオントークンを使用して、OpenDocument で認証できます。

<content>

RESTful 応答のペイロード。<content> 要素には <attrs> 要素が含まれ、それにはさらに一連の <attr> 要素が含まれます。

```
<content>
  <attrs>
    <attr>...</attr>
    <attr>...</attr>
  </attrs>
</content>
```

<attrs>

コンテンツのプロパティのリスト。<attrs> 要素には、一連の <attr> 要素が含まれます。

```
<attrs>
  <attr>...</attr>
  <attr>...</attr>
</attrs>
```

<attr>

コンテンツのプロパティ。

各 <attr> 要素は、コンテンツのプロパティを定義します。<attr> タグは 2 つの属性を使用します。name 属性はプロパティの名前を示し、type 属性はプロパティのタイプを示します。次の例では、コンテンツの id プロパティが値 43 (整数) で、コンテンツの name プロパティが Application Folder (文字列) です。

```
<attr name="id" type="int32">43</attr>
<attr name="name" type="string">Application Folder</attr>
```

次の表では、<attr> タグの name 属性と type 属性に指定できる値を説明します。

名前	種類	説明
name	文字列	リソースの名前。
id	int32	リソースの ID。
cuid	文字列	リソースの一意の ID。
type	文字列	Folder、InfoView などのリソースのタイプ。
description	文字列	リソースの説明。
logonToken	文字列	ログオントークン。

<error>

エラーコード。

<error_code> 要素は RESTful Web Services エラーコード参照を teh 形式 RWS 000xx で示し、<message> 要素には簡潔な説明が含まれます。詳細については、BusinessObjects XI のエラー メッセージの説明ガイドを参照してください。

3.1.5 応答本文から JSON 形式への変換

Business Intelligence プラットフォーム RESTful Web サービス SDK は、JSON 形式の応答と要求ヘッダー `accept : application/json` を提供します。このセクションでは、RESTful Web サービスに JSON タグがどのように適用されるかを説明します。

{ ... }

JSON オブジェクトは、中かっこの { と } で囲みます。これは、XML `<feed>` 要素と同じです。

```
{
  "__metadata": {
    "uri": "http://commandcom-lcm:6405/biprws/infostore/4/children?
page=1&pageSize=50"
  },
  "first": {
    "__deferred": {
      "uri": "http://commandcom-lcm:6405/biprws/infostore/4/children?
page=1&pageSize=50"
    }
  },
  "last": {
    "__deferred": {
      "uri": "http://commandcom-lcm:6405/biprws/infostore/4/children?
page=1&pageSize=50"
    }
  },
  "entries":
  [
    {
      "__metadata": {
        "uri": "commandcom-lcm:6405/biprws/infostore/Alert%20Notifications"
      },
      "id": 64,
      "cuid": "ARZB.BFCQk9PqagDpcFwolw",
      "name": "Alert Notifications",
      "type": "Folder",
      "uri": "alslsls"
    }
    .
    .
    .
  ]
}
```

"entries":

エントリとは、配列内の JSON オブジェクトです。形式は、`"entries" : [{contentsOfEntryItem#1}, {contentsOfEntryItem#2}]` です。次の例は、`../infostore` RESTful Web サービス API 要求の結果です。応答の `"entries":` 部分には、`"Alert Notifications"` と `"Users"` という名前の 2 つの子が表示されます。

```
"entries":
[
```

```

{
  "__metadata": {
    "uri": "commandcom-lcm:6405/biprws/infostore/Alert%20Notifications"
  },
  "id": 64,
  "cuid": "ARZB.BFCQk9PqaqDpcFwo1w",
  "name": "Alert Notifications",
  "type": "Folder",
  "uri": "alslsls"
},
.
.
.
{
  "__metadata": {
    "uri": "http://commandcom-lcm:6405/biprws/infostore/Users"
  },
  "id": 19,
  "cuid": "AXhmigik4CBKra9ZYzR2ezE",
  "description": "",
  "name": "Users",
  "type": "Folder"
}
}

```

__metadata: { uri:

__metadata: { uri: 要素は、XML <link> 要素と同じです。これは、他の RESTful Web サービス要求で利用できる URL へのリンクを定義します。これには、親フォルダ、子フォルダ、要求に関連する追加情報などを入れることができます。これらのリンクをたどることで、BI platform リポジトリを参照できます。

リンクタグの href 属性はハイパーリンクを定義し、rel 属性はリンクのタイプを示します。次のリストでは、rel 属性に指定できる値を説明します。JSON には、XML タグの alternate や related と同等のタグはありません。

属性	形式	例	説明
self	__metadata: { uri:	"__metadata": { "uri": "http:// localhost:6405/ biprws/infostore/4/ children? page=1&pageSize=5" }	この URL へのリンク。
first	first: { __deferred: { uri:	"first": { "__deferred": { "uri": "http:// localhost:6405/ biprws/infostore/4/ children? page=1&pageSize=5" }	結果の最初のページへのリンク。
next	next: { __deferred: { uri:	"next": { "__deferred": { "uri": "http:// localhost:6405/ biprws/infostore/4/	結果の次のページへのリンク。

属性	形式	例	説明
		children? page=2&pageSize=5"}}	
previous	previous: { __deferred: { uri:	"previous": { "__deferred": { "uri": "http:// localhost:6405/ biprws/infostore/4/ children? page=6&pageSize=5"}}	結果の前のページへのリンク。
last	last: { __deferred: { uri:	"last": { "__deferred": { "uri": "http:// localhost:6405/ biprws/infostore/4/ children? page=7&pageSize=5"}}	結果の最後のページへのリンク。
up	up: { __deferred: { uri:	"up": { "__deferred": { "uri": "http:// localhost:6405/ biprws/infostore"}}	現在のリソースの親へのリンク。
children	children: { __deferred: { uri:	"Children": { "__deferred": { "uri": "http:// localhost:6405/ biprws/infostore/ User%20Folders/ children"}}	現在のリソースの子へのリンク。
openDocument	opendocument { __deferred: { uri:	"openDocument": { "__deferred": { "uri": "http:// commandcom-lcm: 8080/BOE/ OpenDocument/ opendoc/ openDocument.jsp? sIDType=CUID&iDocID= AQtkbbSqN4NOj3ydf.Sw 11Y"}}	OpenDocument でリソースを表示するためのリンク。
スケジュール	schedule { __deferred: { uri:	"Scheduling forms": { "__deferred": { "uri": "http:// localhost:6405/ biprws/infostore/ 4930/scheduleForms"}}	リソースをスケジュールするためのリンク。テンプレートを取得するには Get を使用し、要求を送信するには Post を使用します。
		Post を使用し、次のようにスケジュールを挿入します。	

属性	形式	例	説明
		<pre>"__metadata": { "uri": "http://localhost:6405/biprws/infostore/4930/scheduleForms/hourly" }</pre>	

たとえば、次のリンク要素は、結果の最後のページへのリンクを示します。

```
"last": {
  "__deferred": {
    "uri": "http://commandcom-lcm:6405/biprws/infostore/4/children?page=1&pageSize=50"
  }
}
```

Web Intelligence や Crystal Reports などのドキュメントの種類を含む応答からは、電子メールで送信したリポートのボタンコントロールにアタッチできる openDocument URL も得られます。

次の例では、../infostore API を使用して、Web Intelligence openDocument 形式のリンクの一覧を取得します。

```
http://commandcom-lcm:6405/biprws/infostore/4930
```

```
{
  "up": {
    "__deferred": {
      "uri": "http://10.162.204.68:6405/biprws/infostore/4904"
    }
  },
  "Scheduling forms": {
    "__deferred": {
      "uri": "http://10.162.204.68:6405/biprws/infostore/4907/scheduleForms"
    }
  },
  "id": 4907,
  "cuid": "AQtkbbSqN4NOj3ydf.Sw11Y",
  "openDocument": {
    "__deferred": {
      "uri": "http://commandcom-lcm:8080/BOE/OpenDocument/opensdoc/openDocument.jsp?sIDType=CUID&iDocID=AQtkbbSqN4NOj3ydf.Sw11Y"
    }
  },
  "description": "",
  "name": "Formatting Sample",
  "type": "Webi"
}
```

OpenDocument の詳細については、*OpenDocument* によるドキュメントの表示を参照してください。

i 注記

../logon/long API を使用すると、openDocument URL に追加できるログオントークン文字列を取得できるため、受信者はログオン認証情報を提供する必要がありません。

エントリ プロパティ

いくつかのプロパティにより、各エントリ項目のコンテンツが作成されます。次の例は、コンテンツの `id` プロパティが値 64 (整数) で、コンテンツの `name` プロパティが `Alert Notifications` (文字列) であることを示しています。

```
{
  "__metadata": {"uri": "commandcom-lcm:6405/biprws/infostore/Alert
%20Notifications"},
  "id": 64,
  "cuid": "ARZB.BFCQk9PqaqDpcFwolw",
  "name": "Alert Notifications",
  "type": "Folder",
  "uri": "alslsls"
},
```

次の表では、JSON エントリで有効な `name` プロパティと `type` プロパティについて説明します。

名前	種類	例	説明
<code>name</code>	文字列	<code>"name": "Alert Notifications"</code>	リソースの名前。
<code>id</code>	int32	<code>"id": 64</code>	リソースの ID 番号。
<code>cuid</code>	文字列	<code>"cuid": "ARZB.BFCQk9PqaqDpcFwolw"</code>	リソースの一意の ID。
<code>type</code>	文字列	<code>"type": "Folder"</code>	Folder、InfoView などのリソースのタイプ。
<code>description</code>	文字列	<code>"description": "Contains the ..."</code>	リソースの説明。
<code>logonToken</code>	文字列	<code>"type": "COMMANDCOM-LCM:6400@{3&...Sv3b6vUJZe9...}"</code>	ログオントークン。
<code>uri</code>	文字列	<code>"uri": "http://localhost:6405/biprws/infostore/Custom%20Roles"</code>	URI 値。
<code>openDocument</code>	文字列	<code>"openDocument": {"__deferred": {"uri": "http://commandcom-lcm:8080/BOE/OpenDocument/.opendoc/openDocument.jsp?sIDType=CUID&iDocID=AQtkbbSqN4NOj3ydf.Sw11Y"}}</code>	openDocument 形式の URI 値。

error_code

error_code 要素と message 要素はそれぞれ、BI platform エラーまたは RESTful Web サービス エラー("RWS" がプレフィックス)を参照し、簡潔な説明を格納しています。詳細については、SAP BusinessObjects XI のエラー メッセージの説明ガイドを参照してください。

```
{
  "error_code": "FWM 01003",
  "message": "Server COMMANDC-OM-LCM:6400 not found or server may be down (FWM 01003)"
}
```

JSON エスケープ文字

RESTful Web サービスは、JSON が特殊文字と見なす ASCII 文字の前にバックスラッシュ(¥)を付けて返します。文字をエスケープする必要がある JSON の仕様は、<http://www.ietf.org/rfc/rfc4627.txt> にあります。次の表に、RESTful Web サービスの JSON 要求が前にバックスラッシュを付けて返すいくつかの一般的な ASCII++ 文字をリストします。

RWS - JSON	Unicode UTF-8	説明
¥b	U+0008	Backspace
¥f	U+000C	改ページ
¥n	U+000A	新規行
¥r	U+000D	改行
¥t	U+0009	タブ
¥v	U+000B	垂直タブ
¥'	U+0027	単一引用符
¥"	U+0022	二重引用符
¥¥	U+005C	バックスラッシュまたは逆斜線
¥/	U+005D	スラッシュまたは斜線
¥u	U+xxxx	four-hex-digits

3.1.6 XML と JSON の属性の比較

XML を使用する RESTful Web Services 要求は、Atom 仕様に準拠した一部のデータを常に返します。以下の XML タグは、JSON データ形式に同等の要素がありません。このため、容易に認識することができます。

- <author>
- <id>
- <title>

- <updated>
- <link rel=alternate>
- <link rel=related>
- <content>
- <attrs>

サポートされている XML タグと JSON オブジェクト

次の表に、XML タグ、および BI platform RESTful Web サービス実装でサポートされている同等の JSON オブジェクトとエントリをリストします。

表 2: サポートされている XML タグと JSON オブジェクト

XML			JSON		説明
XML タグ	サンプル	型	値	種類	
<feed>			{	JSON オブジェクト	JSON の結果では、応答は JSON オブジェクトとして表されます。XML の <feed> タグは、JSON の最も外側の中かっこ ({})と同じです。
<entry>			entries : [{contentsOfEntryItem#1}, {contentsOfEntryItem#2}]		子のリストの要求。エントリのコレクションが返されます。各コレクションは JSON オブジェクトです。JSON オブジェクトのコレクションは、「エントリ」名と値のペアによる配列として表されます。
<author>			JSON に同等の要素はありません		JSON では、これらの要素は公開されません。
<id>					
<title>					
<updated>					
<link>	rel=self		__metadata: { uri:		現在のロケーションへのリンク。
	rel=first		first: { __deferred: { uri::		結果の最初のページへのリンク。
	rel=next		next: { __deferred: { uri::		結果の次のページへのリンク。
	rel=previous		previous: { __deferred: { uri::		結果の前のページへのリンク。
	rel=last		last: { __deferred: { uri::		結果の最後のページへのリンク。
	rel=alternate		JSON に同等の要素はありません。		現在のロケーションへの別のリンク。
	rel=up		up: { __deferred: { uri::		現在のリソースの親へのリンク。
	rel=related		JSON に同等の要素はありません。		関連リソースへのリンク。
	rel=http://www.sap.com/ rws/ bip#children		children: { __deferred: { uri::		現在のリソースの子へのリンク。

XML			JSON		説明
XML タグ	サンプル	型	値	種類	
	rel=http://www.sap.com/ rws/ bip#opendocument		opendocument: { __deferred: { uri::		レポートや Adobe Acrobat PDF ファイルなどのドキュメントを開く際に使用できるリンク。
	rel=http://www.sap.com/ rws/ bip#schedule		schedule: { __deferred: { uri::		リソースをスケジュールするためのリンク。
<content>			JSON に同等の要素はありません。		XML のみ。これは、<attrs> 要素のコンテナです。Atom フィード仕様には <content> が必要ですが、JSON には必要ありません。
<attrs>					1 つ以上の <attr> 要素を含む XML 要素。JSON では、属性は XML の <attrs> タグのようにグループ化されるのではなく、リソースを表す JSON オブジェクト内に直接、名前と値のペアで表されます。
<attr>	name=name	文字列	name:	JSON 文字列	リソースの名前。
	name=id	int32	id:	JSON 番号	リソースの数値での ID 番号。
	name=cuid	文字列	cuid:	JSON 文字列	クラスターで一意な ID (23 文字の英数字)。
	name=type	文字列	type:	JSON 文字列	Folder、InfoView などのリソースのタイプ。
	name=description	文字列	description:	JSON 文字列	リソースの説明。
	name=logonToken	文字列	logonToken:	JSON 文字列	ログオントークンの文字列。

例

/infostore 要求で取得される XML と JSON 形式の比較

次のコード スニペットは、一般的な /infostore GET 要求で取得される RESTful Web サービス要素の階層を示しています。左側は XML でのリストです。右側は、同じ要求から取得される JSON でのリストです。比較しやすいように、対応する情報の行を左右に配置しています。コード スニペットの長さを短くするために、「Alert Notifications (アラート通知)」と呼ばれる最初のオブジェクトのみを示しています。このスクリーンショットには、前述の表に示されている使用可能なタグの一部が含まれていないことに注意してください。

XML	JSON
<pre> <?xml version="1.0" ?> <feed xmlns="http://www.w3.org/2005/Atom"> <id>tag:sap.com,2010:bip-rs/infostore</id> <title type="text">InfoStore (@COM...CM:6400)</title> <updated>2012-01-13T20:47:42.942Z</updated> <link href="http://...?page=1&pageSize=5" rel="self"/> <link href="http://...?page=1&pageSize=5" rel="first"/> <link href="http://...?page=6&pageSize=5" rel="previous"/> <link href="http://...?page=7&pageSize=5" rel="last"/> <entry> <title type="text">Alert Notifications</title> <id>tag:sap.com,2010:bip-rsARZB.BF...aqDpcFwo1w</id> <author><name>System Account</name></author> <link href="...infostore/Alert%20Notifications" rel="alternate"/> <content type="application/xml"> <attrs xmlns="http://www.sap.com/rws/bip"> <attr name="id" type="int32">64</attr> <attr name="cuid" type="string">ARZB.BF...pcFwo1w</attr> <attr name="description" null="true" type="string"/> <attr name="type" type="string">Folder</attr> </attrs> </content> </entry> </feed> </pre>	<pre> { "__metadata": { "uri": "http://...?page=1&pageSize=5", "first": { "__deferred": { "uri": "http://...?page=1&pageSize=5" } }, "previous": { "__deferred": { "uri": "http://...?page=6&pageSize=5" } }, "last": { "__deferred": { "uri": "http://...?page=7&pageSize=5" } } }, "entries": [{ "name": "Alert Notifications", "__metadata": { "uri": "http://...infostore/Alert%20Notifications", "id": 64, "cuid": "ARZB.BF...aqDpcFwo1w", "type": "Folder" } }] } </pre>

3.1.7 多言語データの操作

多言語環境では、コンテンツとシステムメッセージを優先言語で返すように要求できます。コンテンツとシステムメッセージの優先言語の定義に使用される要求ヘッダー属性には、Accept-Language と X-SAP-PVL の 2 つがあります。

BI platform ソフトウェアがインストールされると、ユーザーインターフェイスとシステム エラー メッセージは、製品ロケール (PL) で表示されます。利用できる PL 言語としては、BI platform ソフトウェアと共にインストールされる言語パックがあります。

エラーメッセージなどのシステムメッセージは、PL で指定される言語で返されます。Accept-Language 要求ヘッダー属性を設定することで、システムメッセージに特定の言語を使用するように要求できます。たとえば、システムメッセージを日本語で取得するには、Accept-Language 要求ヘッダー属性を ja-JP に設定します。

i 注記

要求した PL がない場合は、BI platform ソフトウェアのインストール時に使用された PL でシステムメッセージが返されます。

BI platform のコンテンツは多言語で保存できます。たとえば、BI platform は、フランス語、日本語、およびドイツ語に翻訳されたレポートを保存できます。返されるコンテンツの優先言語を指定するには、X-SAP-PVL 要求ヘッダー属性を使用します。

要求された言語のコンテンツがない場合は、最も近い言語でコンテンツが返されます。たとえば、フランス語のコンテンツを要求する場合は、X-SAP-PVL 要求ヘッダー属性を fr-FR に設定します。

HTML 言語コードの詳細については、HTML 4.01 仕様(<http://www.w3.org>)を参照してください。

3.2 RESTful Web サービス要求の取得

Business Intelligence プラットフォーム RESTful Web サービス SDK を使用するには、RESTful Web サービス要求をリスニングするサービスのプロトコル、サーバー名、ポート番号、およびパスの情報がが必要です。これらを組み合わせてベース URL が形成されます。RESTful Web サービスへの要求は常に、最初にベース URL があり、その後に具体的な要求内容が続きます。

BI platform 単一のサーバーにインストールされる基本インストールでは、デフォルトのベース URL として `http://<servername>:6405/biprws/` が使用されます。

複雑なデプロイメントシナリオでは、RESTful Web サービスをホストする Web アプリケーションコンテナサーバー(WACS)のインスタンスが複数ある場合があります。その場合は、RESTful Web サービスを別の場所でホストすることもできます。BI platform 管理者は RESTful Web サービスへのアクセスに使用する場所のベース URL を定義し、ユーザーはプログラムまたはセントラル管理コンソール(CMC)を使用してこのベース URL を調べることができます。

3.2.1 CMC によるベース URL の取得

RESTful Web サービス要求のベース URL を調べるには、セントラル管理コンソール(CMC)ユーザーインターフェイスにログインし、RESTful Web サービス設定に移動します。

1. CMC にログインします。
2. [アプリケーション]をクリックします。
3. [RESTful Web サービス]を右クリックし、[プロパティ]をクリックします。
[RESTful Web サービス]プロパティウィンドウが表示されます。
4. [アクセス URL]テキストボックスからベース URL を取得します。

3.2.2 プログラムによるベース URL の取得

BI platform Java SDK などの他の BI platform SDK の 1 つを使用して、RESTful Web サービスのベース URL をプログラムで調べることができます。RESTful Web サービスのベース URL をプログラムで調べるには、まず、BI platform に照会して RESTful Web サービスオブジェクトの `SI_ACCESS_URL` プロパティを取得する必要があります。RESTful Web サービスオブジェクトは、その CUID または種類に基づいて照会できます。CUID と種類は、Java 定数の `com.businessobjects.sdk.plugin.desktop.restwebservice.IRestWebService.CUID` と `com.businessobjects.sdk.plugin.desktop.restwebservice.IRestWebService.KIND` にアクセスして調べることができます。

i 注記

RESTful Web サービスの CUID 値は AZpJlb9HDtxPjLHwEmF8xD8、種類の値は RestWebService です。

```
"SELECT SI_ACCESS_URL FROM CI_APOBJECTS WHERE SI_CUID='" + IRestWebService.CUID +  
"'"
```

```
"SELECT SI_ACCESS_URL FROM CI_APOBJECTS WHERE SI_KIND='" + IRestWebService.KIND +  
"'"
```

BI platform Java SDK バージョン 4.1 によるベース URL の検出

IRestWebService インターフェイスの getURL メソッドを使用して、RESTful Web サービスのベース URL を取得することができます。

```
IInfoObjects objects= infostore.query("SELECT SI_ACCESS_URL FROM CI_APOBJECTS  
WHERE SI_CUID='" + IRestWebService.CUID + "'");  
IInfoObject object = (IInfoObject)objects.get(0);  
IRestWebService restAppObject = (IRestWebService) object;  
String baseUrl = restAppObject.getURL();
```

BI platform Java SDK の詳細については、*SAP BusinessObjects Business Intelligence プラットフォーム Java SDK 開発者ガイド*を参照してください。

3.3 認証

Business Intelligence プラットフォーム RESTful Web サービス SDK から BI platform にアクセスするには、ログオントークンが必要です。ログオン URL に要求を行ってログオントークンを取得します。このトークンは、有効なユーザーとして認証されていることを証明します。また、以降の RESTful Web サービス要求にこのトークンを含めることで、パスワードなどの機密情報をささす必要がなくなります。

以下の情報タイプの 1 つを使用して、認証およびその結果となるログオントークンを取得することができます。

- BI platform ログオン認証情報。この方法は、WinAD、LDAP、SAP、Enterprise の各認証をサポートします。認証の詳細については、*SAP BusinessObjects Business Intelligence プラットフォーム管理者ガイド*を参照してください。
- 別のセッションのセッショントークン。既に認証されているセッションへのアクセス権を持つ場合は、そのセッショントークンを使用してログオントークンを取得できます。

i 注記

別の SDK から取得したセッショントークンは、ログオントークンと同じではなく、RESTful Web サービス要求で直接使用できません。

- シリアライズセッション。既に認証されているセッションへのアクセス権を持つ場合は、それを使用してログオントークンを取得できます。

認証要求が成功した場合は、応答ヘッダーにログオントークンが挿入されます。このログオントークンは、X-SAP-LogonToken によって定義されます。

i 注記

応答本文には、ログオントークンのコピーが含まれます。ただし、このログオントークンのコピーは XML に埋め込まれており、&、<、> などの不正な XML 文字は、XML 適合形式に変換(エンコード)されています。このログオントークンのコピーを使用するには、XML エンコード文字を元の形式に変換する必要があります。または、応答ヘッダーに含まれるログオントークンのコピーを使用することもできます。これは XML 形式に変換されていません。

RESTful Web サービスに要求を行うたびに要求ヘッダーに X-SAP-LogonToken 属性を追加し、その値には、認証の際に受け取ったログオントークンを設定する必要があります。ログオントークンには、要求ヘッダー内で使用できない文字が含まれていることがあるため、ログオントークンは引用符で囲んでください。

以下の表は、ログオントークンの例です。

属性	サンプル値
X-SAP-LogonToken	"COMMANDCOM-LCM: 6400@{3&2=5604,U3&p=40623.9446463889,Y7&4F =12,U3&63=secEnterprise, 0P&68=secEnterprise:Administrator, 0P&qe=100,U3&vz=g5KUV8cAA.d_ARmSDnBy6T7jJV NyFCTso4s0q3dI.4k,UP}"

関連情報

[XML エンコードテキストからログオントークンへの変換 \[37 ページ\]](#)

3.3.1 ユーザー名とパスワードからログオントークンを取得する

BI platform にログオンするには、RESTful Web サービス要求のベース URL を取得してある必要があります。

BI platform にログオンし、ログオントークンを取得するには、要求本文にユーザー名、パスワード、および認証タイプを指定し、POST メソッドを使用して `http://<baseURL>/logon/long` に要求を行います。

以下のタイプの認証を使用して、BI platform にログオンすることができます。

- WinAD
- LDAP
- SAP
- エンタープライズ

ログオン要求の本文をフォーマットする方法を調べるには、GET メソッドを使用して、同じ URL `http://<baseURL>/logon/long` に要求を行います。この応答には、ログオン要求の要求本文のフォーマットに使用できる XML テンプレートが含まれます。この XML テンプレートには、サポートされる認証タイプのリストが含まれます。

1. 新しい HTTP 要求を作成します。
2. 要求ヘッダーに Accept 属性を追加し、その値を application/xml に設定します。
3. GET メソッドを使用して、http://<baseURL>/logon/long URL に要求を送信します。

<baseURL> は、RESTful Web サービスのベース URL に置き換えてください。

```
GET http://localhost:6405/biprws/logon/long
```

応答本文にはテンプレートが含まれます。

```
<attrs xmlns="http://www.sap.com/rws/bip">
  <attr name="userName" type="string"/></attr>
  <attr name="password" type="string"/></attr>
  <attr name="auth" type="string"
possibilities="secEnterprise,secLDAP,secWinAD,secSAPR3">secEnterprise</attr>
</attrs>
```

4. 新しい HTTP 要求を作成します。
5. 要求ヘッダに Accept 属性を追加し、その値を application/xml に設定します。
6. 要求ヘッダーに Content-Type 属性を追加し、その値を application/xml に設定します。
7. XML テンプレートにユーザー名、パスワード、および認証タイプを設定し、それを新しい要求の要求本文に追加します。

```
<attrs xmlns="http://www.sap.com/rws/bip">
  <attr name="userName" type="string">myUserName</attr>
  <attr name="password" type="string">myPassword</attr>
  <attr name="auth" type="string"
possibilities="secEnterprise,secLDAP,secWinAD,secSAPR3">secEnterprise</attr>
</attrs>
```

8. POST メソッドを使用して、同じ URL http://<baseURL>/logon/long に要求を送信します。

<baseURL> は、RESTful Web サービスのベース URL に置き換えてください。

```
POST http://localhost:6405/biprws/logon/long
```

応答ヘッダーは、ログオントークンを X-SAP-LogonToken 属性として返します。

```
X-SAP-LogonToken:"COMMANDCOM-LCM:
6400@{3&2=5595,U3&p=40674.9596541551,Y7&4F=12,U3&63=secEnterprise,
0P&66=60,03&68=secEnterprise:Administrator,
0P&qe=100,U3&vz=SFY6agrLPxpfQBK1ZKYCwoBZKCbfsQm7VgWZFiH.RhM,UP"
```

ログオントークンは引用符に囲われます。上記の例では、ログオントークンは以下のとおりです。

```
COMMANDCOM-LCM:6400@{3&2=5595,U3&p=40674.9596541551,Y7&4F=12,U3&63=secEnterprise,
0P&66=60,03&68=secEnterprise:Administrator,
0P&qe=100,U3&vz=SFY6agrLPxpfQBK1ZKYCwoBZKCbfsQm7VgWZFiH.RhM,UP
```

応答本文には、<attr> 要素にログオントークンのコピーが含まれます。ログオントークンに XML では不正な文字が含まれている場合、それらは XML エンコード値に置き換えられます。たとえば、& 文字は & に置き換えられます。応答本文から取得したログオントークンを使用するには、XML エンコードログオントークンを元の形式に変換する必要があります。

次の例は、応答本文に表示される XML エンコードログオントークンを示します。

```
<entry xmlns="http://www.w3.org/2005/Atom">
  <author><name>@COMMANDCOM-LCM:6400</name></author>
```

```
<id>tag:sap.com,2010:bip-rs/logon/long</id>
<title type="text">Logon Result</title>
<updated>2011-03-07T20:48:56.015Z</updated>
<content type="application/xml">
  <attrs xmlns="http://www.sap.com/rws/bip">
    <attr name="logonToken" type="string">COMMANDCOM-LCM:6400@{3&
2=5595,U3&p=40674.9596541551,Y7&4F=12,U3&63=secEnterprise,0P&
66=60,03&68=secEnterprise:Administrator,
0P&qe=100,U3&vz=SFY6agrLPxpfQBK1ZKYCwoBZKCbfsQm7VgWZFiH.RhM,UP}</attr>
  </attrs>
</content>
</entry>
```

関連情報

[RESTful Web サービス要求の取得 \[28 ページ\]](#)

[XML エンコードテキストからログオントークンへの変換 \[37 ページ\]](#)

3.3.2 シリアライズセッションまたはセッショントークンからログオントークンを取得する

この方法でログオンするには、BI platform Java SDK などの別の BI platform SDK を使用して、既存の認証済みセッションにアクセスする必要があります。また、RESTful Web サービス要求のベース URL 情報が必要です。

RESTful Web サービスのログオントークンは、有効なセッショントークンまたはシリアライズセッションから取得できます。POST メソッドを使用して、`http://<baseURL>/logon/token` URL に要求を行い、XML エンコードバージョンのシリアライズセッションまたはセッショントークンを要求本文に指定します。<baseURL> は、RESTful Web サービスのベース URL に置き換えてください。

要求本文をフォーマットする方法を調べるには、GET メソッドを使用して、同じ URL `http://<baseURL>/logon/token` に要求を行います。この要求の応答には、ログオン要求の要求本文で利用できる XML テンプレートが含まれます。

シリアライズセッションを使用してログオントークンを取得すると、BI platform によって使用される同時接続ユーザーライセンスの数が増加しません。一方、セッショントークンを使用すると、同時接続ユーザーライセンスの数が 1 つ増加します。

1. 新しい HTTP 要求を作成します。
2. GET メソッドを使用して、`http://<baseURL>/logon/token` URL に要求を送信します。

<baseURL> は、RESTful Web サービスのベース URL に置き換えてください。

```
GET http://localhost:6405/biprws/logon/token
```

応答には、XML テンプレートが含まれます。

```
<attrs xmlns="http://www.sap.com/rws/bip">
  <attr name="tokenType" type="string" possibilities="token,
serializedSession">token</attr>
  <attr name="logonToken" type="string" null="true"></attr>
</attrs>
```

3. 新しい HTTP 要求を作成します。

4. 要求ヘッダーに Content-Type 属性を追加し、その値を application/xml に設定します。
5. XML テンプレートに入力し、それを要求本文に追加します。

セッショントークンを使用している場合は <attr name="tokenType" type="string"> 要素の値を token に設定し、シリアライズセッションを使用している場合はそれを serializedSession に設定します。<attr name="logonToken" type="string"> 要素の値を XML エンコードバージョンのシリアライズセッションまたはセッショントークンの値に設定します。

```
<attrs xmlns="http://www.sap.com/rws/bip">
  <attr name="tokenType" type="string" possibilities="token,
serializedSession">serializedSession</attr>
  <attr name="logonToken" type="string">3&ua=AWmaEx4Z.NVPpAEthuTGAjc,
8P&ub=AfRWaT5_131NlLLf5bRMLKY,8P&S5,88&
5U=5320JaqlNvF1mr4m8u5UQFadItj5319JWKkfBwlKLBfrgXC8NpgljC,8P&
63=secEnterprise,8P&2r=COMMANDCOM-LCM:6400,8P&3k=@COMMANDCOM-LCM:
6400,8P&1=Administrator account,8P&W={},?z&
4E=5319JWKkfBwlKLBfrgXC8NpgljC,8P&Tn={3&.1={3&
2=726,03&O=FavoritesFolder,0P},2z&.2={3&
2=727,03&O=PersonalCategory,0P},2z&.3={3&2=728,03&O=Inbox,0P},
2z&U=3,03},?z&4F=12,8P&Tm=36500,83&uy=-1043,8L&
35=Administrator,8P&ux=AeiCInd_R6lBrV98duvX1dc,8P&pa,8P</attr>
</attrs>
```

i 注記

ここでは、シリアライズセッションの例を示します。シリアライズセッションまたはセッショントークン値は、XML エンコードして、不正な XML 文字を削除する必要があります。たとえば、& 文字は & に置き換えてください。

6. POST メソッドを使用して、同じ URL `http://<baseURL>/logon/token` に要求を送信します。<baseURL> は、RESTful Web サービスのベース URL に置き換えてください。

```
POST http://localhost:6405/biprws/logon/token
```

応答ヘッダーは、ログオントークンを X-SAP-LogonToken 属性として返します。

```
X-SAP-LogonToken:"COMMANDCOM-LCM:
6400@{3&2=5595,U3&p=40674.9596541551,Y7&4F=12,U3&63=secEnterprise,
0P&66=60,03&68=secEnterprise:Administrator,
0P&qe=100,U3&vz=SFY6agrLPxpfQBK1ZKYCwoBZKCbfsQm7VgWZFiH.RhM,UP"
```

ログオントークンは引用符に囲われます。

i 注記

応答本文には、<attr> 要素にログオントークンのコピーが含まれます。ログオントークンに XML では不正な文字が含まれている場合、それらは XML エンコード値に置き換えられます。たとえば、& 文字は & に置き換えられます。応答本文から取得したログオントークンを使用するには、XML エンコードログオントークンを元の形式に変換する必要があります。

次の例は、応答本文に表示される XML エンコードログオントークンを示します。

```
<entry xmlns="http://www.w3.org/2005/Atom">
  <author><name>@COMMANDCOM-LCM:6400</name></author>
  <id>tag:sap.com,2010:bip-rs/logon/token</id>
  <title type="text">Logon Result</title>
  <updated>2011-06-28T17:54:31.994Z</updated>
  <content type="application/xml">
    <attrs xmlns="http://www.sap.com/rws/bip">
```

```
<attr name="logonToken" type="string">COMMANDCOM-LCM:6400@{3&#x219,U3&#x2D;40722.7462034491,Y7&#x2D;4F=12,U3&#x2D;63=secEnterprise,0P&#x2D;66=60,03&#x2D;68=secEnterprise:Administrator,0P&#x2D;qe=100,U3&#x2D;vz=KeDu7064jWSptBT_m5BkBj5Q_NaxyvE_WStqXmigYrg,UP}</attr>
</attrs>
</content>
</entry>
```

関連情報

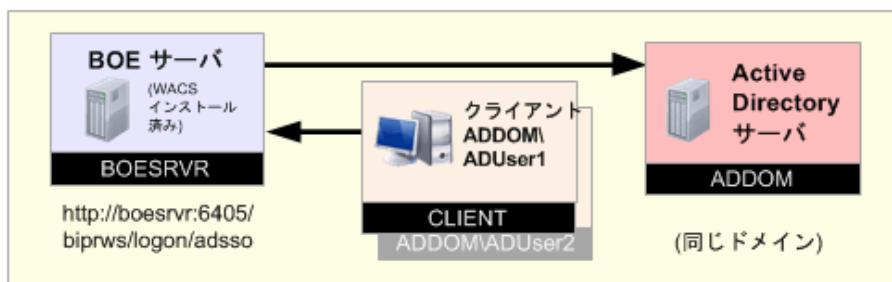
[RESTful Web サービス要求の取得 \[28 ページ\]](#)

[他の SDK から取得した認証済みセッションの使用 \[41 ページ\]](#)

[XML エンコードテキストからログオントークンへの変換 \[37 ページ\]](#)

3.3.3 Active Directory シングル サインオン(AD SSO)アカウントを使用してログオントークンを取得する

RESTful Web サービスの Active Directory シングル サインオン機能を使用するには、クライアントに Windows Active Directory (WinAD) アカウントが必要です。また、クライアントは、/logon/adsso API を使用するコンピュータにログインする必要があります。また、クライアントには、WinAD アカウントに対応するログオン アカウントが BI platform に必要です。次の図に、BI platform サーバー、クライアントコンピュータ、および Windows Active Directory サーバーの間の設定と認証の関係を示します。



クライアントは、「管理タスクとインストール タスク > WinAD SSO を有効化するように web.xml を設定する」の手順に従って WinAD SSO 機能を有効にすると、自身の WinAD 認証情報を使用して、コンピュータにログオンできます。この認証情報は、BI platform サーバーに自動的にアクセスする際の認証にも使用されます。

次の手順に従って、AD SSO でログオントークンを取得します。

1. 新しい HTTP 要求を作成します。
2. GET メソッドを使用して、`http://<baseURL>/logon/adsso` に要求を送信します。

<baseURL> は、RESTful Web サービスのベース URL に置き換えてください。

例:

```
GET http://localhost:6405/biprws/logon/adsso
```

応答ヘッダーは、ログオントークンを X-SAP-LogonToken 属性として返します。次のような XML 応答例が表示されます。

```
<?xml version="1.0" ?>
<entry xmlns="http://www.w3.org/2005/Atom">
  <author>
    <name>
      @BOESRVR.ADDOM.COM
    </name>
  </author>
  <id>
    tag:sap.com,2010:bip-rs/logon/adsso
  </id>
  <title type="text">
    Logon Result
  </title>
  <updated>
    2011-11-11T11:11:11.340Z
  </updated>
  <content type="application/xml">
    <attrs xmlns="http://www.sap.com/rws/bip">
      <attr name="logonToken" type="string">
        BOESRVR.ADDOM.COM:6400@{3&2=4584,U3&p=40868.9276775116,Y7&4F=4331,U3
        &63=secWinAD,0P&66=60,03&68=secWinAD:CN%3DADUser1%2CCN%3DUsers%2CDC%3D
        ADDOM%2CDC%3DCOM,0P&qe=100,U3&vz=
        kOox8TDqAiFsfs8T3GefI3sWXIyKymc9qvvtAjihC7w,UP}
      </attr>
    </attrs>
  </content>
</entry>
```

3. 結果として得られる HTTP 要求ヘッダ内の X-SAP-LogonToken を使用して、さらに RESTful Web サービス要求 (例: `http://<baseURL>/infostore`) を作成します。また、ログオントークンを HTTP エンコードして、`&token=<logonToken>` パラメータと共に OpenDocument URL に付加することができます。

関連情報

[RESTful Web サービス要求の取得 \[28 ページ\]](#)

[XML エンコードテキストからログオントークンへの変換 \[37 ページ\]](#)

3.3.4 信頼できる認証を使用して、ログオントークンを取得する

RESTful Web サービスの信頼できる認証機能を使用するには、「管理タスクとインストール タスク」>「信頼できる認証を有効にして構成する」の説明に従って、この機能を有効にする必要があります。

ユーザーが別の場所で認証されている場合、信頼できる認証を使用すると、保護されたリソースへのアクセス速度が向上します。たとえば、ユーザーが Windows アカウントを使用してログインしている場合などです。

次の方法で、信頼できる認証を使用するログオントークンを取得できます。

- ユーザー名にカスタマイズ可能なヘッダーを使用する HTTP ヘッダー要求。
- URL クエリー。
- Cookie 認証。

これらのいずれかの方法で、信頼できる認証を使用するログオントークンを取得するには、CMC を開いて[取得方法]メニューの[WACS]>[信頼できる認証設定]に移動し、使用する方法に合うようにオプションを変更します。信頼できる認証を使用するログオントークンを取得する際に、[名前パラメータ]を変更できます。このオプションは、[サーバー]>[コアサービス]>[WACS]にあります。提供されているすべての URL と値で、大文字と小文字が区別されます。

取得方法	使用する RESTful API	使用方法
HTTP_HEADER	/logon/trusted	<ol style="list-style-type: none"> 1. GET メソッドを使用して、HTTP 要求を作成します。 2. /logon/trusted API を使用します。たとえば、<code>http://localhost:6405/biprws/logon/trusted</code> のようにします。 3. デフォルトラベル X-SAP-TRUSTED-USER を含む要求ヘッダーを作成し、bob などの信用できるユーザー名を追加します。最終的なログオントークンが応答ヘッダーに表示されます。
QUERY_STRING	/logon/trusted?<MyUser>=<Username>	<ol style="list-style-type: none"> 1. Web ブラウザで、/logon/trusted API を使用してユーザー名パラメータとユーザー名を URL に追加します。たとえば、<code>http://localhost:6405/biprws/logon/trusted?MyUser=bob</code> のようにします。たとえば、次のようになります。 <ul style="list-style-type: none"> ◦ カスタマイズされたユーザ名でパラメータ MyUser を置き換えます。このパラメータは、CMC のサーバー > コアサービス > WACS > 信頼できる認証設定で設定します。 ◦ 信用できるユーザーの名前で bob を置き換えます。信用できるユーザーは、CMC の[ユーザーとグループ]>[ユーザー一覧]で設定します。 最終的なログオントークンがブラウザの表示ウィンドウに表示されます。
COOKIE	/logon/trusted	<ol style="list-style-type: none"> 1. Cookie を作成し、以下の情報を追加します。 <ul style="list-style-type: none"> ◦ ドメイン。localhost などです。

取得方法	使用する RESTful API	使用方法
		<ul style="list-style-type: none"> 名前ラベル。たとえば、デフォルト値の X-SAP-TRUSTED-USER とログオン名の値 (bob など) です。 パス (/ (スラッシュ) など)。 <p>2. http://localhost:6405/biprws/logon/trustedなどの URL を入力し、Enter キーを押すと、ブラウザ ウィンドウに最終的なログオントークンが表示されます。</p>

関連情報

[RESTful Web サービス要求の取得 \[28 ページ\]](#)

[XML エンコードテキストからログオントークンへの変換 \[37 ページ\]](#)

3.3.5 XML エンコードテキストからログオントークンへの変換

ログオントークンは、認証応答の応答ヘッダーと応答本文の両方に入れて返されます。応答本文は XML 形式ですが、XML ではいくつかの文字が独自の目的で使用するために予約されています。ログオントークンにこのような文字が含まれている場合は、XML に埋め込み可能な文字シーケンスに置き換えられますが、それらはログオントークンの中では機能しません。XML エンコードされたログオントークンを使用するには、元の形式に変換する必要があります。

i 注記

この手順は、応答本文からログオントークンを取得する場合にのみ必要です。応答ヘッダーに格納されているログオントークンは、XML エンコードされていません。

XML エンコードされたログオントークンを元の形式に変換するには、XML エンコードされた各文字シーケンスを本来の文字に置き換えます。たとえば、`&` 文字エンコードは `&` 文字に置き換えます。

次の表に、許可されない XML 文字を XML エンコードする一般的な例を示します。

XML エンコード	文字
<code>&apos;</code>	'
<code>&quot;</code>	"
<code>&amp;</code>	&
<code>&lt;</code>	<
<code>&gt;</code>	>

XML で表示される文字の詳細については、<http://www.w3.org> で拡張マークアップ言語の仕様を参照してください。

例

これは、XML エンコードされたログオントークンの例です。

```
COMMANDCOM-LCM:6400@{3&2=5675,U3&p=40653.0083064583,Y7&4F=12,U3&63=secEnterprise,0P&66=60,03&68=secEnterprise:Administrator,0P&qe=100,U3&vz=y3EqvsvoehahHhbmPrpaPjKVMU8raN3zEpnt2YjqDe4,UP}
```

これは、元の形式に変換されたログオントークンです。

```
COMMANDCOM-LCM:6400@{3&2=5675,U3&p=40653.0083064583,Y7&4F=12,U3&63=secEnterprise,0P&66=60,03&68=secEnterprise:Administrator,0P&qe=100,U3&vz=y3EqvsvoehahHhbmPrpaPjKVMU8raN3zEpnt2YjqDe4,UP}
```

3.3.6 要求ヘッダーにログオントークンを追加する

ログオントークンを取得したら、それを使用して、BI platform にアクセスする RESTful 要求を認証することができます。

i 注記

要求本文からログオントークンを取得した場合は、それを XML エンコード形式から元の形式に変換する必要があります。または、応答ヘッダーから元のログオントークンを直接取得することもできます。

たとえば、このテキストは、応答本文の XML に埋め込まれたログオントークンです。

```
COMMANDCOM-LCM:6400@{3&2=5675,U3&p=40653.0083064583,Y7&4F=12,U3&63=secEnterprise,0P&66=60,03&68=secEnterprise:Administrator,0P&qe=100,U3&vz=y3EqvsvoehahHhbmPrpaPjKVMU8raN3zEpnt2YjqDe4,UP}
```

このテキストは、応答ヘッダーから取得したログオントークン、または元の形式に変換された応答本文から取得したトークンです。

```
COMMANDCOM-LCM:6400@{3&2=5675,U3&p=40653.0083064583,Y7&4F=12,U3&63=secEnterprise,0P&66=60,03&68=secEnterprise:Administrator,0P&qe=100,U3&vz=y3EqvsvoehahHhbmPrpaPjKVMU8raN3zEpnt2YjqDe4,UP}
```

1. 新しい RESTful Web サービス要求を作成するか、既存の要求を変更します。
2. 要求ヘッダーに属性を追加します。
3. 属性の名前を X-SAP-LogonToken に設定します。
4. 属性の値をログオントークンの値に設定し、その値を引用符で囲みます。

名前	値
X-SAP-LogonToken	"COMMANDCOM-LCM:6400@{3&2=5604,U3&p=40623.9456463889,Y7&4F=12,U3&63=secEnterprise,0P&68=secEnterprise:Administrator,0P&qe=100,U3&vz=g5KUU8cAA.d_ArMSDnBy6T7jJVNYFCTso4s0q3dI.4k,UP}"

関連情報

[XML エンコードテキストからログオントークンへの変換 \[37 ページ\]](#)

3.3.7 HTTP 基本認証の使用

HTTP 基本認証を使用すると、RESTful Web サービス要求の HTTP ヘッダーにログオントークンを含めることなく、BI platform にログオンできます。その代わりに、ユーザー名、パスワード、および認証タイプを指定します。

i 注記

HTTPS と組み合わせて使用しない限り、HTTP 基本認証のユーザー名とパスワードは安全に転送されません。

HTTP 基本認証は、管理者が有効にする必要があります。また、管理者は、ユーザーが認証タイプを指定しない場合に使用されるデフォルトの認証タイプを定義することもできます。

認証タイプ

HTTP 基本認証では、以下の認証タイプを使用できます。

- secEnterprise - Enterprise 認証
- secLDAP - LDAP 認証
- secWinAD - Windows AD 認証
- secSAPR3 - SAP 認証

HTTP 認証を使用して要求を行うと、ライセンスが 1 つ使用されます。セッションキャッシュを使用しない場合は、要求の持続期間中 1 つのライセンスが使用され、要求が完了するとそれが解放されます。セッションキャッシュを使用する場合は、キャッシュされたセッションに関連付けられたライセンスが使用されます。

i 注記

ユーザー名、パスワード、および認証タイプは、RFC 2716 で定義されている base64 エンコードである必要があります。HTTP 基本認証では、: 文字を含むユーザ名を使用できません。

Web ブラウザーでの HTTP 基本認証の使用

デフォルトの認証タイプを使用して Web ブラウザーでログオンするには、プロンプトでユーザー名とパスワードを指定します。

特定の認証タイプを使用してログオンするには、ユーザー名フィールドで <authenticationType>¥<username> を使用し、パスワードプロンプトでパスワードを指定します。<authenticationType> は認証タイプに、<username> はユーザー名に置き換えてください。たとえば、ユーザー名 myUserName を使用して SAP 認証でログオンするには、ユーザー名フィールドに secSAPR3¥myUserName と入力し、パスワードフィールドにパスワードを入力します。

i 注記

セッションは複数のブラウザ要求間で保持されません。基本認証によって実行した各サービス要求で、base64 形式でエンコードされた依頼ヘッダのユーザ名およびパスワードが必要です。

プログラムによる HTTP 基本認証の使用

プログラムから HTTP 基本認証を使用するには、要求ヘッダーに `Authorization` 属性を追加し、その値として認証文字列の base64 エンコード値を設定します。

デフォルトの認証タイプを使用するには、次の認証文字列を使用します。

```
Basic <username>:<password>
```

特定の認証タイプを使用するには、次の認証文字列を使用します。

```
Basic <authtype>%<username>:<password>
```

3.3.8 BI platform からログオフする

BI platform からログオフするには、RESTful Web サービス要求のベース URL 情報が必要です。また、無効化するセッションのログオントークンが必要です。

ログオントークンは、指定した時間使用されないと、自動的に期限切れになります。デフォルトでは、1 時間アイドル状態が続くとログオントークンが期限切れになりますが、管理者はこの値を構成することができます。ログオントークンが自動的に期限切れになる前に、セッションをログオフするには、`http://<baseURL>/logoff` URL に POST 要求を行います。

<baseURL> は、RESTful Web サービスのベース URL に置き換えてください。

BI platform からログオフすることで、ログオントークンが無効化され、セッションに関連付けられたライセンスが解放されます。

1. 新しい HTTP 要求を作成します。
2. 要求ヘッダーに `Accept` 属性を追加し、その値を `application/xml` に設定します。
3. 要求ヘッダーに `X-SAP-LogonToken` 属性を追加し、その値に引用符で囲んだログオントークン値を設定します。

名前	値
X-SAP-LogonToken	"COMMANDCOM-LCM: 6400@{3&2=5604,U3&p=40623.9456463889,Y7& 4F=12,U3&63=secEnterprise, 0P&68=secEnterprise:Administrator, 0P&qe=100,U3&vz=g5KUU8cAA.d_ARmSDnBy6T7j JVNYFCTso4s0q3dI.4k,UP}"

4. POST メソッドを使用して、`http://<baseURL>/logoff` URL に要求を送信します。

<baseUrl> は、RESTful Web サービスのベース URL に置き換えてください。

```
POST http://<baseUrl>/logout
```

ログオフが成功すると、応答ヘッダーの HTTP ステータスコードに 200 が設定されます。

関連情報

[RESTful Web サービス要求の取得 \[28 ページ\]](#)

3.3.9 他の SDK から取得した認証済みセッションの使用

別の BI platform SDK を使用して、既存の認証済みセッションからシリアル化セッションまたはセッショントークンを取得できます。次に、このシリアル化セッションまたはセッショントークンを /login/token URL への要求に指定して、Business Intelligence プラットフォーム RESTful Web サービス SDK のログオントークンを取得できます。

以下の SDK バージョン XI 3.0 以上から取得したシリアル化セッションまたはセッショントークンを使用できます。

- SAP BusinessObjects Business Intelligence プラットフォーム Java SDK
- SAP BusinessObjects Business Intelligence プラットフォーム .NET SDK
- SAP BusinessObjects Business Intelligence プラットフォーム Web サービス SDK

3.3.9.1 BI platform Java SDK によるセッション情報の取得

BI platform Java SDK を使用して、既に認証されている既存のセッションからシリアル化セッションまたはセッショントークンを取得できます。/login/token URL への要求の本文にシリアル化セッションまたはセッショントークンを指定して、Business Intelligence プラットフォーム RESTful Web サービス SDK 用のログオントークンを取得します。

シリアル化セッションを取得するには、IEnterpriseSession クラスの getSerializedSession メソッドを使用します。

```
ISessionMgr sessionMgr = CrystalEnterprise.getSessionMgr();
IEnterpriseSession enterpriseSession = sessionMgr.logon("username", "password",
"cmsname", "secEnterprise");
String serializedSession = enterpriseSession.getSerializedSession();
```

セッショントークンを取得するには、ILogonTokenMgr クラスの getDefaultToken メソッドまたは createLogonToken メソッドを使用します。

```
ISessionMgr sessionMgr = CrystalEnterprise.getSessionMgr();
IEnterpriseSession enterpriseSession = sessionMgr.logon("username", "password",
"cmsname", "secEnterprise");
String sessionToken = enterpriseSession.getLogonTokenMgr().getDefaultToken();
```

BI platform Java SDK の使用の詳細については、SAP BusinessObjects Business Intelligence プラットフォーム Java SDK 開発者ガイドを参照してください。

3.3.9.2 BI platform .NET SDK によるセッション情報の取得

BI platform .NET SDK を使用して、既に認証されている既存のセッションからシリアル化セッションまたはセッショントークンを取得できます。/login/token URL への要求の本文にシリアル化セッションまたはセッショントークンを指定して、Business Intelligence プラットフォーム RESTful Web サービス SDK 用のログオントークンを取得します。

シリアル化セッションを取得するには、EnterpriseSession クラスの SerializedSession プロパティを使用します。

```
SessionMgr sessionMgr = new SessionMgr();
EnterpriseSession session = sessionMgr.Logon("username", "password", "cms",
"secEnterprise");
string serializedSession = session.SerializedSession;
```

セッショントークンを取得するには、LogonTokenMgr クラスの SerializedSession プロパティまたは CreateLogonTokenEx メソッドを使用します。

```
SessionMgr sessionMgr = new SessionMgr();
EnterpriseSession session = sessionMgr.Logon("username", "password", "cms",
"secEnterprise");
string logonTokenMgr = session.LogonTokenMgr.DefaultToken;
```

3.3.9.3 BI platform Web サービス SDK によるセッション情報の取得

BI platform Web サービス SDK を使用して、既に認証されている既存のセッションからシリアル化セッションまたはセッショントークンを取得できます。/login/token URL への要求の本文にシリアル化セッションまたはセッショントークンを指定して、Business Intelligence プラットフォーム RESTful Web サービス SDK 用のログオントークンを取得します。

シリアル化セッションを取得するには、SessionInfo クラスの getSerializedSession メソッドを使用します。

```
URL boConURL = new URL("http://boserver:port/dswsbobje/services/Session");
Connection connection = new Connection(boConURL);
Session session = new Session(connection);
EnterpriseCredential credential = EnterpriseCredential.Factory.newInstance();
credential.setLogin("username");
credential.setPassword("password");
credential.setDomain("domain");
credential.setAuthType("secEnterprise");
SessionInfo sessionInfo = session.login(credential);
String serializedSession = sessionInfo.getSerializedSession();
```

セッショントークンを取得するには、SessionInfo クラスの getDefaultToken メソッドを使用します。

```
URL boConURL = new URL("http://boserver:port/dswsbobje/services/Session");
Connection connection = new Connection(boConURL);
Session session = new Session(connection);
EnterpriseCredential credential = EnterpriseCredential.Factory.newInstance();
credential.setLogin("username");
credential.setPassword("password");
credential.setDomain("domain");
credential.setAuthType("secEnterprise");
SessionInfo sessionInfo = session.login(credential);
String sessionToken = sessionInfo.getDefaultToken();
```

BI platform Web サービスコンシューマ Java SDK の使用の詳細については、*SAP BusinessObjects Business Intelligence プラットフォーム Web サービスコンシューマ Java SDK 開発者ガイド*を参照してください。

3.4 レポート ID の取得

RESTful Web サービスを使用して Crystal レポートにアクセスするには、レポート ID が必要です。レポート ID は、プログラムまたはセントラル管理コンソール(CMC)で調べることができます。CMC でレポートを右クリックすると、レポート ID がレポートプロパティと共に表示されます。

BI platform リポジトリのナビゲートの詳細については、*Business Intelligence Platform RESTful Web サービス開発者ガイド*を参照してください。

3.4.1 プログラムからレポート ID を取得する

有効なログオントークンが必要です。また、レポートが BI platform リポジトリのどこに保存されているかが既知である必要があります。

レポート ID を取得するには、RESTful Web サービスを使用してリポジトリを参照します。

1. 新しい HTTP 要求を作成します。
2. 要求ヘッダに X-SAP-LogonToken 属性を追加し、その値を有効なログオントークンに設定します。
3. GET メソッドを使用して、`http://<baseURI>/infostore/Root%20Folder/children` URL に要求を送信します
このメソッドは、BI platform リポジトリのルートディレクトリにあるコンテンツの概要を返します。要求応答でフォルダ ID が返されます。この応答例では、Report Samples フォルダの ID は 5375 です。

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <id>tag:sap.com,2010:bip-rs/ASHnC0S_Pw5LhKFbZ.iA_j4/children</id>
  <title type="text">Children of Root Folder</title>
  <updated>2011-05-30T20:02:41.470Z</updated>
  ...
  <entry>
    <title type="text">Report Samples</title>
    <id>tag:sap.com,2010:bip-rs/AfwlNA25oGlFr8gN2TxHddA</id>
    <author>
      <name>Administrator</name>
      <uri>http://localhost:6405/biprws/infostore/12</uri>
    </author>
    <link href="http://localhost:6405/biprws/infostore/5375" rel="alternate"></link>
    <content type="application/xml">
      <attrs xmlns="http://www.sap.com/rws/bip">
        <attr name="cuid" type="string">AfwlNA25oGlFr8gN2TxHddA</attr>
        <attr name="description" type="string">Contains sample reports.</attr>
        <attr name="type" type="string">Folder</attr></attrs></content>
    </entry>
    ...
  </feed>
```

4. 新しい HTTP 要求を作成します。
5. 要求ヘッダに X-SAP-LogonToken 属性を追加し、その値を有効なログオントークンに設定します。
6. GET メソッドを使用して、`http://<baseURI>/infostore/<folderID>/children/` URL に要求を送信します。

i 注記

<folderID> は、フォルダ ID に置き換えてください。

このメソッドは、フォルダコンテンツの概要と、対応するオブジェクト ID を返します。名前を基準にしてレポートがリストされ、コンテンツ概要と共にレポート ID が提供されます。この応答例では、Drilldown レポートの ID は 5567 です。

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <id>tag:sap.com,2010:bip-rs/AYLus319xBdAh0yC3SCxU6I/children</id>
  <title type="text">Children of Feature Samples</title>
  ...
  <entry>
    <title type="text">Drilldown</title>
    <id>tag:sap.com,2010:bip-rs/AbHK3o8Bnz1MpOXtKWa_1V8</id>
    <author>
      <name>Administrator</name>
      <uri>http://localhost:6405/biprws/infostore/12</uri>
    </author>
    <link href="http://localhost:6405/biprws/infostore/5567" rel="alternate"></link>
    <content type="application/xml">
      <attrs xmlns="http://www.sap.com/rws/bip">
        <attr name="cuid" type="string">AbHK3o8Bnz1MpOXtKWa_1V8</attr>
        <attr name="description" type="string"></attr>
        <attr name="type" type="string">CrystalReport</attr>
      </attrs>
    </content>
  </entry>
  ...
</feed>
```

ステップ 4 ~ 6 を繰り返して、ネストしたフォルダを順に参照します。

関連情報

[レポートの URI \[46 ページ\]](#)

4 SAP Crystal Reports RESTful Web サービス API の使用

RESTful Web サービス API は、正確には既存のレポート内のデータを操作するためのツールです。新しい SAP Crystal Reports を作成することはできません。

SAP Crystal Reports RESTful Web サービスの動作方法は、RESTful に準拠します。GET を使用してレポートを取得し、POST を使用して一時レポートインスタンスを変更することができます。取得した結果はユーザが読み取り可能で、余分なマークアップは含まれていません。

RESTful Web サービス要求はステートレスです。サーバが作成する通常のレポートジョブは、1 時間のアイドル期間の後に再利用されます。最近サーバに要求されたレポートを要求すると、既存のレポートジョブが共有されることもあります。

i 注記

PUT、UPDATE、DELETE、MERGE、HEAD、OPTIONS の各メソッドはサポートされません。

4.1 共通 URI

よく使用される URI:

- http://<baseURI>
- http://<baseURI>/infostore/<reportID>/rpt
- http://<baseURI>/infostore/<reportID>/rpt/export?mime_type=application/pdf
- http://<baseURI>/infostore/<reportID>/rpt/data.svc
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows(<INDEX>)
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows(<INDEX>)/<PROPERTY>
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows(<INDEX>)/<PROPERTY>/<VALUE>
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows?\$skip=<VALUE>
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows?\$skiptoken=<VALUE>
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows?\$inlinecount=allpages
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows/\$count
- http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows?<PARAMETER>=<VALUE>
- http://<baseURI>/infostore/<reportID>/rpt/GrandTotals
- http://<baseURI>/infostore/<reportID>/rpt/instance

関連情報

[RESTful Web サービス要求の取得 \[28 ページ\]](#)

[レポート ID の取得 \[43 ページ\]](#)

4.2 レポートの URI

レポート情報の取得と変換を行います。

i 注記

デフォルトの表現形式は XML です。

URI	操作	サポートされる形式	AtomPub か
<baseURI>/infostore/ <reportID>/rpt	GET	XML、JSON	いいえ

例

GET <http://<baseURI>/infostore/<reportID>/rpt/>

レポートの要約情報を取得するための GET 要求を行います。

要求:

- メソッド: GET
- URL: <http://<baseURI>/infostore/<reportID>/rpt/>
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答:

- レポートの要約情報を含むエントリが返されます。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<report_summary_info xmlns="http://schemas.sap.com/crystalreports/
report_summary_info">
  <report_name>DateAndTime.rpt</report_name>
  <report_title></report_title>
  <file_format_version>
    <major_version>12</major_version>
    <minor_version>0</minor_version>
  </file_format_version>
  <author></author>
  <subject></subject>
  <keywords></keywords>
  <comments></comments>
  <last_data_refresh_date>2010-12-17T11:59:51.000</last_data_refresh_date>
  <export_uri>http://localhost:6405/biprws/infostore/6231/rpt/export</
export_uri>
  <service_uri>http://localhost:6405/biprws/infostore/6231/rpt/data.svc</
service_uri>
  <edit_uri>http://localhost:6405/biprws/infostore/6231/rpt/instance</edit_uri>
```

```
</report_summary_info>
```

4.3 レポートメタデータ

この API を使用して、Crystal レポートからメタデータ構造を取得することができます。

i 注記

デフォルトの表現形式は XML です。

URI	操作	サポートされる形式	AtomPub か
<baseURI>/infostore/ <reportID>/rpt/ structure	GET	XML、JSON	×

例

GET <http://<baseURI>/infostore/<reportID>/rpt/structure>

レポートの要約情報を取得するための GET 要求を行います。

要求:

- メソッド: GET
- URI: <http://<baseURI>/infostore/<reportID>/rpt/structure>
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答:

- 応答には以下に示すようなレポートの概要が含まれます。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<ns17:reportstructure>
  <datasources>
    <datasource>
      <datasourcename>10.160.197.218_Latest - Oracle12c_JDBC_Connection_10</
datasourcename>
      <connection>
        <cuid>AUy28zmbGthPtalH6n3Gshg</cuid>
        <kind>SQL</kind>
        <networklayer>JDBC</networklayer>
        <database>Oracle 12</database>
      </connection>
      <tables>
        <table>
          <name>CUSTOMER</name>
          <tablekind>SQL_Table</tablekind>
          <dbfields>
            <dbfield>
              <name>CUST_ID</name>
              <kind>SQL_Field</kind>
              <datatype>decimal</datatype>
```

```

<technicalname>CUST_ID</technicalname>
<qualifiedname>CUSTOMER.CUST_ID</qualifiedname>
</dbfield>
<dbfield>
<name>FIRST_NAME</name>
<kind>SQL_Field</kind>
<datatype>string</datatype>
<technicalname>FIRST_NAME</technicalname>
<qualifiedname>CUSTOMER.FIRST_NAME</qualifiedname>
</dbfield>
<dbfield>
<name>LAST_NAME</name>
<kind>SQL_Field</kind>
<datatype>string</datatype>
<technicalname>LAST_NAME</technicalname>
<qualifiedname>CUSTOMER.LAST_NAME</qualifiedname>
</dbfield>
<dbfield>
<name>AGE</name>
<kind>SQL_Field</kind>
<datatype>decimal</datatype>
<technicalname>AGE</technicalname>
<qualifiedname>CUSTOMER.AGE</qualifiedname>
</dbfield>
<dbfield>
<name>PHONE_NUMBER</name>
<kind>SQL_Field</kind>
<datatype>string</datatype>
<technicalname>PHONE_NUMBER</technicalname>
<qualifiedname>CUSTOMER.PHONE_NUMBER</qualifiedname>
</dbfield>
<dbfield>
<name>ADDRESS</name>
<kind>SQL_Field</kind>
<datatype>string</datatype>
<technicalname>ADDRESS</technicalname>
<qualifiedname>CUSTOMER.ADDRESS</qualifiedname>
</dbfield>
<dbfield>
<name>CITY_ID</name>
<kind>SQL_Field</kind>
<datatype>decimal</datatype>
<technicalname>CITY_ID</technicalname>
<qualifiedname>CUSTOMER.CITY_ID</qualifiedname>
</dbfield>
<dbfield>
<name>SALES_ID</name>
<kind>SQL_Field</kind>
<datatype>decimal</datatype>
<technicalname>SALES_ID</technicalname>
<qualifiedname>CUSTOMER.SALES_ID</qualifiedname>
</dbfield>
<dbfield>
<name>SPONSOR_ID</name>
<kind>SQL_Field</kind>
<datatype>decimal</datatype>
<technicalname>SPONSOR_ID</technicalname>
<qualifiedname>CUSTOMER.SPONSOR_ID</qualifiedname>
</dbfield>
</dbfields>
</table>
<table>
<name>REGION</name>
<tablekind>SQL_Table</tablekind>
<dbfields>
<dbfield>
<name>REGION_ID</name>
<kind>SQL_Field</kind>

```



```

<datatype>decimal</datatype>
<technicalname>REGION_ID</technicalname>
<qualifiedname>REGION.REGION_ID</qualifiedname>
</dbfield>
<dbfield>
<name>REGION</name>
<kind>SQL_Field</kind>
<datatype>string</datatype>
<technicalname>REGION</technicalname>
<qualifiedname>REGION.REGION</qualifiedname>
</dbfield>
<dbfield>
<name>COUNTRY_ID</name>
<kind>SQL_Field</kind>
<datatype>decimal</datatype>
<technicalname>COUNTRY_ID</technicalname>
<qualifiedname>REGION.COUNTRY_ID</qualifiedname>
</dbfield>
</dbfields>
</table>
<table>
<name>COUNTRY</name>
<tablekind>SQL_Table</tablekind>
<dbfields>
<dbfield>
<name>COUNTRY_ID</name>
<kind>SQL_Field</kind>
<datatype>decimal</datatype>
<technicalname>COUNTRY_ID</technicalname>
<qualifiedname>COUNTRY.COUNTRY_ID</qualifiedname>
</dbfield>
<dbfield>
<name>COUNTRY</name>
<kind>SQL_Field</kind>
<datatype>string</datatype>
<technicalname>COUNTRY</technicalname>
<qualifiedname>COUNTRY.COUNTRY</qualifiedname>
</dbfield>
</dbfields>
</table>
<table>
<name>CITY</name>
<tablekind>SQL_Table</tablekind>
<dbfields>
<dbfield>
<name>CITY_ID</name>
<kind>SQL_Field</kind>
<datatype>decimal</datatype>
<technicalname>CITY_ID</technicalname>
<qualifiedname>CITY.CITY_ID</qualifiedname>
</dbfield>
<dbfield>
<name>CITY</name>
<kind>SQL_Field</kind>
<datatype>string</datatype>
<technicalname>CITY</technicalname>
<qualifiedname>CITY.CITY</qualifiedname>
</dbfield>
<dbfield>
<name>REGION_ID</name>
<kind>SQL_Field</kind>
<datatype>decimal</datatype>
<technicalname>REGION_ID</technicalname>
<qualifiedname>CITY.REGION_ID</qualifiedname>
</dbfield>
</dbfields>
</table>
</tables>

```

```

</datasource>
<joins>
<join>
<sourcetable>REGION</sourcetable>
<kind>Inner</kind>
<targettable>COUNTRY</targettable>
<fieldlinks>
<fieldlink>
<fromfield>
<name>COUNTRY_ID</name>
<kind>DatabaseField</kind>
<datatype>decimal</datatype>
</fromfield>
<fieldlinkOperator>Equal</fieldlinkOperator>
<tofield>
<name>COUNTRY_ID</name>
<kind>DatabaseField</kind>
<datatype>decimal</datatype>
</tofield>
</fieldlink>
</fieldlinks>
</join>
<join>
<sourcetable>REGION</sourcetable>
<kind>Inner</kind>
<targettable>CITY</targettable>
<fieldlinks>
<fieldlink>
<fromfield>
<name>REGION_ID</name>
<kind>DatabaseField</kind>
<datatype>decimal</datatype>
</fromfield>
<fieldlinkOperator>Equal</fieldlinkOperator>
<tofield>
<name>REGION_ID</name>
<kind>DatabaseField</kind>
<datatype>decimal</datatype>
</tofield>
</fieldlink>
</fieldlinks>
</join>
<join>
<sourcetable>CITY</sourcetable>
<kind>Inner</kind>
<targettable>CUSTOMER</targettable>
<fieldlinks>
<fieldlink>
<fromfield>
<name>CITY_ID</name>
<kind>DatabaseField</kind>
<datatype>decimal</datatype>
</fromfield>
<fieldlinkOperator>Equal</fieldlinkOperator>
<tofield>
<name>CITY_ID</name>
<kind>DatabaseField</kind>
<datatype>decimal</datatype>
</tofield>
</fieldlink>
</fieldlinks>
</join>
</joins>
</datasources>
<usedfields>
<usedfield fieldtype="DatabaseField" uniqueid="{CUSTOMER.CITY_ID}"
name="CITY_ID" />
<usedfield fieldtype="DatabaseField" uniqueid="{CITY.CITY}" name="CITY" />

```

```

<usedfield fieldtype="DatabaseField" uniqueid="{COUNTRY.COUNTRY}"
name="COUNTRY" />
<usedfield fieldtype="DatabaseField" uniqueid="{CUSTOMER.FIRST_NAME}"
name="FIRST_NAME" />
<usedfield fieldtype="DatabaseField" uniqueid="{CUSTOMER.LAST_NAME}"
name="LAST_NAME" />
<usedfield fieldtype="DatabaseField" uniqueid="{REGION.REGION}"
name="REGION" />
</usedfields>
<subreports>
<subreportstructureuri>http://10.160.199.155:6405/biprws/infostore/6518/rpt/
First+SubReport/structure</subreportstructureuri>
<subreportstructureuri>http://10.160.199.155:6405/biprws/infostore/6518/rpt/
Second%26%21%40%23%24%25%5E%26*%28%29SubReport/structure</
subreportstructureuri>
</subreports>
</ns17:reportstructure>

```

表 3:

タグ名	説明
<ns17:reportstructure>	<p>これにはレポート構造が含まれます。サブタグは以下のとおりです。</p> <ul style="list-style-type: none"> • <datasources> • <usedfields> • <groupconditions> • <parameters> • <summary> • <runningtotals> • <subreports>
<datasources>	<p>レポートで使用するデータソースの一覧が含まれます。</p> <p>これには <datasource> がサブタグとして含まれます。</p>
<datasource>	<p>これにはレポートで使用する特定のデータソースに関する情報が含まれます。このタグには以下のサブタグが含まれます。</p> <ul style="list-style-type: none"> • <datasourcename>: これはデータソースの名前です。 • <connection>: これにはデータソース接続の詳細が含まれます。 • <tables>: これにはテーブルの詳細が含まれます。 <p>i 注記</p> <p>各 <datasource> タグは 1 つの接続を示します。</p>

タグ名	説明
<connection>	<p>これにはデータソース接続の詳細が含まれます。</p> <p>これには以下のサブタグが含まれます。</p> <ul style="list-style-type: none"> • <cuid>: これは一意の接続 ID です。 • <kind>: これは以下のような接続のタイプです。 <ul style="list-style-type: none"> ◦ SQL - リレーション接続 ◦ DSL - その他の接続 • <database>: これには以下のタイプのデータベースが含まれます。 <ul style="list-style-type: none"> ◦ PQUERY - 分析ビュー ◦ BICS - BICS 接続 ◦ DSL - ユニバース ◦ HANA-HANA OLAP 接続 ◦ <DATABASE> - 接続されたデータベースサーバ名とバージョンが表示されます • <datasourcelanguagelevel>: データソースの言語レベル
<joins>	<p>これには単一または複数の結合情報が含まれます。</p> <p>これには <join> がサブタグとして含まれます。</p>
<join>	<p>これにはテーブル間の結合情報が含まれます。</p> <ul style="list-style-type: none"> • <sourcetable>: これにはソーステーブル名が含まれます。 • <kind>: これには結合のタイプが含まれます。 • <targettable>: これにはターゲットテーブル名が含まれます。 • <fieldlinks>: これにはフィールドリンク情報が含まれます。
<fieldlinks>	<p>これにはフィールドリンク情報が含まれます。これには 1 つの <fieldlink> のみがサブタグとして含まれます。</p>
<fieldlink>	<p>これには以下のサブタグが含まれます。</p> <ul style="list-style-type: none"> • <fromfield> : これにはソーステーブルのフィールド情報が含まれます。 • <fieldlinkOperator>: これには Equal、NotEqual、LessThan などの演算子名が含まれます。 • <tofield> : これには出力先テーブルのフィールド情報が含まれます。

タグ名	説明
<fromfield>	<p>これにはソーステーブルのフィールド情報が含まれます。これには以下のサブタグが含まれます。</p> <ul style="list-style-type: none"> • <name>: これはフィールド名です。 • <kind>: これはフィールドタイプです。
<tofield>	<p>これには出力先テーブルのフィールド情報が含まれます。</p> <ul style="list-style-type: none"> • <name>: これは出力先テーブルのフィールド名です。 • <kind>: これは出力先テーブルのフィールドタイプです。
<tables>	<p>これにはテーブルの情報が含まれます。これには table がサブタグとして含まれます。</p>
<table>	<p>これにはテーブル固有の情報が含まれ、以下のサブタグが含まれます。</p> <ul style="list-style-type: none"> • <name>: テーブル名 • <tablekind>: 以下のようなテーブルタイプ: <ul style="list-style-type: none"> ◦ SQL_Table: リレーション接続 ◦ Universe_Table: その他の接続 • <dbfields>: データベースフィールド情報
<dbfields>	<p>これにはデータベースフィールド情報が含まれます。これには <dbfield> がサブタグとして含まれます。</p>
<dbfield>	<p>このタグには以下のサブタグが含まれます。</p> <ul style="list-style-type: none"> • <name>: データベースフィールド名 • <kind>: 以下のようなデータベースフィールドタイプ: <ul style="list-style-type: none"> ◦ ディメンション ◦ メジャー ◦ ValueBasedHierarchy • <datatype>: データベースフィールドのデータ型 • <technicalname>: データベースフィールドの技術名 • <qualifiedname>: データベースフィールドの修飾名
<usedfields>	<p>これにはレポートで使用されるフィールドに関する情報が含まれます。</p> <p>このタグには以下の単一または複数のサブタグが含まれます。</p> <p><usedfield>: 使用されるフィールド</p>
<parameters>	<p>このタグには <parameter> または <parametergroup> タグが含まれます。</p> <ul style="list-style-type: none"> • <parameter>: パラメータ情報 • <parametergroup>: パラメータグループ情報

タグ名	説明
<parametergroup>	これには <parameter> がサブタグとして含まれます。
<parameter>	このタグには以下のサブタグが含まれます。 <ul style="list-style-type: none"> • <currentvalues>: パラメータの現在の値 • <initialvalues> : パラメータの初期値
<groupconditions>	これにはグループで使用されるデータベースフィールドが含まれます。 このタグには <groupcondition> がサブタグとして含まれます。
<summary>	これにはレポートで使用される集計フィールドが含まれます。 このタグには <summaryfield> がサブタグとして含まれます。
<runningtotals>	これにはレポートで使用される積算合計が含まれます。 このタグには <runningtotal> がサブタグとして含まれます。
<formulas>	これにはレポートで使用される式が含まれます。 このタグには <formula> がサブタグとして含まれます。
<formula>	これはレポートで使用される式です。 このタグには <formulatext> がサブタグとして含まれます。
<subreports>	これにはサブレポートの URI が含まれます。これを使用してサブレポートの構造にアクセスできます。 このタグには <subreportstructureuri> がサブタグとして含まれます。

4.4 レポートインスタンス

レポートインスタンスは、BI platform リポジトリに保存されているレポートの一時コピーです。GET メソッドは、レポートインスタンスのみを返します。

GET メソッドを使用して、保存データが含まれるレポートのインスタンスを要求すると、RESTful Web サービスは、リポジトリ内でレポートが最後に最新表示されたときに保存されたデータを反映する結果を表示します。これは、要求応答本文の `last_refresh_date` プロパティによって示されます。GET メソッドを使用して、保存データが含まれないレポートのインスタンスを要求すると、レポートのメタデータと要約情報は表示されますが、レポートデータにアクセスする要求は失敗します。

レポートデータは、一時レポートインスタンスにのみ POST することができます。このため、リポジトリに含まれているレポートが RESTful Web サービスによって変更されることはありません。インスタンスから返されるデータは、インスタンスに最後に POST が実行されたときに最新表示されたデータを反映します。

i 注記

一時レポートインスタンスにフィールド名を POST する場合、フィールド名は 480 文字までに制限されます。

保存データがないレポートのデータを表示するには、一時インスタンスを作成する必要があります。

デフォルトでは、一時インスタンスは、1 時間のアイドル状態の後に期限切れになります。サーバが過負荷の状態になると、1 時間が経過する前に、新しいジョブのためにインスタンスが再利用されることもあります。

一時インスタンスを作成するには、/instance URI に対して GET メソッドを使用します。これにより、情報を入力するためのフォームが返されます。POST メソッドを使用して、このフォームを送信する必要があります。このフォームには、接続情報、レポートで使用されるパラメータなどのエントリが含まれます。また、SuppressData エントリも含まれます。これは、デフォルトでは false に設定されます。

- SuppressData が false に設定された場合は、レポートのデータソースから行セットデータが取得されて、一時レポートインスタンスに挿入されます。
- SuppressData が SuppressData に設定された場合は、一時レポートインスタンスにデータが挿入されません。これは、レポートのデータがデータサービスの POST to rows メソッドを使用してプログラムから提供される場合に便利です。

i 注記

このオプションは、ユニバースへの接続なしで作成されたレポートでのみ有効です。

POST メソッドは一時インスタンスを作成し、この新しい一時インスタンスへのインスタンス ID リンクを応答に挿入します。

一時インスタンスを使用するには、インスタンス ID を URI に追加します。例:

```
http://<baseURI>/infostore/<reportID>/rpt/<instanceID>/data.svc/Rows
```

i 注記

<instanceID> は、作成されたインスタンスの ID に置き換えてください。

URI	操作	サポートされる形式	AtomPub か
<baseURI>/infostore/ <reportID>/rpt/ instance	GET、POST	XML	はい

i 注記

日付パラメータまたは時刻パラメータが必要なレポートインスタンスを作成する場合は、Edm.Datetime の規則を使用する必要があります。例: datetime'2010-02-26T17:08:53'。詳細については、対話型パラメータの節を参照してください。

関連情報

[行 \[75 ページ\]](#)

[インタラクティブパラメータ \[65 ページ\]](#)

4.4.1 既存のレポートから一時レポートインスタンスを作成する

1. 新しい HTTP 要求を作成します。
2. GET メソッドを使用して、`http://<baseURI>/infostore/<reportID>/rpt/instance` URL に要求を送信します。

要求の例:

- メソッド: GET
- URL: `http://<baseURI>/infostore/<reportID>/rpt/instance`
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答の例:

- 応答にはフォームが含まれます。

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title type="text">Crystal Reports New Instance</title>
  <entry>
    <title type="text">SuppressData</title>
    <content type="application/xml">
      <attrs>
        <attr name="value" type="boolean">false</attr>
      </attrs>
    </content>
  </entry>
  <entry>
    <title type="text">ConnectionInfo</title>
    <id>reportse--</id>
    <content type="application/xml">
      <attrs>
        <attr name="ServerName" type="string">reportse</attr>
        <attr name="DatabaseName" type="string"></attr>
        <attr name="userName" type="string"></attr>
        <attr name="password" type="string"></attr>
      </attrs>
    </content>
  </entry>
</feed>
```

3. 別の HTTP 要求を作成します。
4. フォームに入力し、POST メソッドを使用して、`http://<baseURI>/infostore/<reportID>/rpt/instance` URL に要求を送信します。

要求の例:

- メソッド: POST
- URL: `http://<baseURI>/infostore/<reportID>/rpt/instance`
- 要求ヘッダ属性: X-SAP-LogonToken, Content-Type
- 要求本文: GET 要求によって取得したフォームに入力します。

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title type="text">Crystal Reports New Instance</title>
  <entry>
    <title type="text">SuppressData</title>
    <content type="application/xml">
      <attrs>
        <attr name="value" type="boolean">false</attr>
      </attrs>
    </content>
  </entry>
</feed>
```



```

    </content>
  </entry>
  <entry>
    <title type="text">ConnectionInfo</title>
    <id>reportse--</id>
    <content type="application/xml">
      <attrs>
        <attr name="ServerName" type="string">reportse</attr>
        <attr name="DatabaseName" type="string">HOSTID</attr>
        <attr name="userName" type="string">USER</attr>
        <attr name="password" type="string">PASSWORD</attr>
      </attrs>
    </content>
  </entry>
</feed>

```

i 注記

<HOSTID> はデータベース名、<USER> はユーザ名、および <PASSWORD> はパスワードに置き換えてください。レポートにパラメータプロンプトが含まれている場合、フォームにはパラメータフィールドも表示されます。

応答の例:

- 応答ヘッダ:

```

HTTP/1.1 201 Created
Content-Type: application/xml
Location: http://localhost:6405/biprws/1667/rpt/eNotTsEKgzAU
        _5r2rBXnduhhqAdhA9GNnUt5KzLbynvdXL.fc_ySkISQ3HReX.
        ntglAtLkm2Op09hiAHXVbVSXZ6FEVjo_ecwDiTgOIzrQaBnck7xkjWMC3GAQF
        _AFkIVSq1TkG9jqh_Yofmuo3dmeMaN0pmHmCJmKjHaIFoCm789w8PXs0Ledmv3HX2BT49NO8
Date: Fri, 27 May 2011 21:52:19 GMT
Content-Length: 513

```

- 応答本文:

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<reportInstance>
  <report_name>DateAndTime.rpt</report_name>
  <resource>http://localhost:6405/biprws/infostore/1667/rpt/eNotTsEKgzAU
        _5r2rBXnduhhqAdhA9GNnUt5KzLbynvdXL.fc_ySkISQ3HReX.
        ntglAtLkm2Op09hiAHXVbVSXZ6FEVjo_ecwDiTgOIzrQaBnck7xkjWMC3GAQF
        _AFkIVSq1TkG9jqh_Yofmuo3dmeMaN0pmHmCJmKjHaIFoCm789w8PXs0Ledmv3HX2BT49NO8</
resource>
  <id>eNotTsEKgzAU_5r2rBXnduhhqAdhA9GNnUt5KzLbynvdXL.
        fc_ySkISQ3HReX.ntglAtLkm2Op09hiAHXVbVSXZ6FEVjo
        _ecwDiTgOIzrQaBnck7xkjWMC3GAQF_AfKIVSq1TkG9jqh
        _Yofmuo3dmeMaN0pmHmCJmKjHaIFoCm789w8PXs0Ledmv3HX2BT49NO8</id>
</reportInstance>

```

応答本文には、新しいインスタンスへの URI、およびそのインスタンスに関連付けられた一意の ID が含まれます。新しいインスタンス ID は、応答ヘッダからも取得できます。

4.5 エクスポート

指定された MIME タイプでレポートをエクスポートします。

URI	操作	サポートされる形式	AtomPub か
<baseURI>/infostore/ <reportID>/rpt/ export? mime_type=<MIMETYPE>	GET		いいえ

i 注記

<MIMETYPE> は、エクスポート先のファイルタイプに置き換えてください。

MIME タイプ	説明
text/csv	文字区切り値 (CSV)
application/vnd.ms-excel	Microsoft Excel (97-2003)
application/vnd.ms-excel&isDataOnly=true	Microsoft Excel データのみ (97-2003)
application/vnd.openxmlformats-officedocument.spreadsheetml.sheet	Microsoft Excel ワークブックデータのみ
application/msword	Microsoft Word
application/msword&isEditable=true	Microsoft Word 編集可能
application/PDF	PDF
application/rtf	リッチテキスト形式 (RTF)
text/ttx	タブ区切りテキスト形式 (TXT)
text/plain	テキスト
application/xml	XML

i 注記

オプションパラメータを使用して、要求をカスタマイズできます。

オプションパラメータ	型	説明
startPageNumber=<VALUE>	整数	<VALUE> は、エクスポートする先頭ページを指定します。
endPageNumber=<VALUE>	整数	<VALUE> は、エクスポートする先頭ページを指定します。

i 注記

開始ページ番号と終了ページ番号を指定しない場合は、すべてのページがエクスポートされます。

例

GET http://<baseURI>/infostore/<reportID>/rpt/export?mime_type=application/PDF

レポートを PDF にエクスポートするための GET 要求を行います。

要求:

- メソッド: GET
- URL: `http://<baseURI>/infostore/<reportID>/rpt/export?mime_type=application/PDF`
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答:

- 応答ヘッダ:

```
Status Code: 200 OK
Server: Apache-Coyote/1.1
Date: Mon 30, May 2011 22:23:22 GMT
Content-Type: application/PDF
Transfer-Encoding: chunked
```

- 応答本文: PDF エンコード形式のレポートが含まれます

4.5.1 文字区切り値(CSV)パラメータ

以下のパラメータを使用して、文字区切り値(CSV)形式でエクスポートするレポートを書式設定できます。

オプションパラメータ	型	説明
?delimiter=<VALUE>	文字列	<VALUE> は、レポートで区切り文字として使用されます。
?separator=<VALUE>	文字列	<VALUE> は、レポートで文字区切りとして使用されます。
?reportSectionsOption=<VALUE>	整数、文字列	<p><VALUE>は、以下の値を使用して、レポートセクションオプションを設定します。</p> <ul style="list-style-type: none">• 0 = Export• 1 = ExportIsolated• 2 = doNotExport <div>i 注記 ベース領域タイプの設定には、整数または文字列のいずれかを使用できます。 例: ?reportSectionsOption="Export"</div>

オプションパラメータ	型	説明
		<p>i 注記</p> <p>他の値を使用すると、例外がスローされます。</p>
? groupSectionsOption=<VALUE>	整数、文字列	<p><VALUE> は、以下の値を使用して、グループセクションオプションを設定します。</p> <ul style="list-style-type: none"> 0 = Export 1 = ExportIsolated 2 = doNotExport <p>i 注記</p> <p>ベース領域タイプの設定には、整数または文字列のいずれかを使用できます。</p> <p>例: ? groupSectionsOption="Export"</p> <p>i 注記</p> <p>他の値を使用すると、例外がスローされます。</p>

4.5.2 Microsoft Excel パラメータ

以下のオプションパラメータを使用して、Microsoft Excel にエクスポートするレポートを書式設定できます。

オプションパラメータ	型	説明
?constantWidth=<VALUE>	整数	<p><VALUE> は、レポートの固定幅を設定します。</p> <p>i 注記</p> <p>constantWidth と baseAreaType は排他的なオプションです。クエリに両方が存在する場合は、固定幅が優先します。</p>
?baseAreaType=<VALUE>	整数、文字列	<p><VALUE> は、以下の値の 1 つを使用して、ベース領域タイプを設定します。</p>

オプションパラメータ	型	説明
		<ul style="list-style-type: none"> • 1 = ReportHeader • 2 = PageHeader • 3 = GroupHeader • 4 = Detail • 5 = GroupFooter • 7 = Page Footer • 8 = ReportFooter • 255 = WholeReport <div> i 注記 ベース領域タイプの設定には、整数または文字列のいずれかを使用できます。 例: ? <code>baseAreaType="ReportHeader"</code> </div> <div> i 注記 他の値を使用すると、例外がスローされます。 </div> <div> i 注記 <code>constantWidth</code> と <code>baseAreaType</code> は排他的なオプションです。クエリに両方が存在する場合は、固定幅が優先します。 </div>
? <code>exportPageHeaderFooter=<VALUE></code>	整数	<p><VALUE> は、以下の値の 1 つを使用して、ページヘッダとページフッタを設定します。</p> <ul style="list-style-type: none"> • 0 = なし • 1 = 1 回 • 2 = 各ページ <div> i 注記 他の値を使用すると、例外がスローされます。 </div>

4.5.3 Microsoft Excel データのみパラメータ

以下のパラメータを使用して、Microsoft Excel データのみ形式にエクスポートするレポートを書式設定したり、そのレポートに関する情報を返すことができます。

オプションパラメータ	型	説明
?constantWidth=VALUE	整数	<p>VALUE は、レポートの幅を設定します。</p> <div>i 注記 constantWidthと baseAreaType は排他的なオプションです。クエリに両方が存在する場合は、固定幅が優先します。</div>
?baseAreaType=VALUE	整数、文字列	<p>VALUE は、以下の値の 1 つを使用して、ベース領域タイプを設定します。</p> <ul style="list-style-type: none">• 1 = ReportHeader• 2 = PageHeader• 3 = GroupHeader• 4 = Detail• 5 = GroupFooter• 7 = Page Footer• 8 = ReportFooter• 255 = WholeReport <div>i 注記 ベース領域タイプの設定には、整数または文字列のいずれかを使用できます。 例: ? baseAreaType="ReportHeader"</div> <div>i 注記 他の値を使用すると、例外がスローされます。</div> <div>i 注記 constantWidthと baseAreaType は排他的なオプションです。クエリに両方が存在する場合は、固定幅が優先します。</div>

オプションパラメータ	型	説明
? exportPageHeaderFooter=VALUE	整数	<p>VALUE は、以下の値の 1 つを使用して、ページヘッダとページフッタをエクスポートします。</p> <ul style="list-style-type: none"> 0 = なし 1 = 1 回 2 = 各ページ <div> i 注記 他の値を使用すると、例外がスローされます。 </div>
?isFormatUsed	論理値	この形式が使用される場合は TRUE、そうでない場合は FALSE を返します。
?isWorksheetFuncUsed	論理値	ワークシート関数が使用される場合は TRUE、そうでない場合は FALSE を返します。
?isColAlignmentMaintained	論理値	列の配置が維持される場合は TRUE、そうでない場合は FALSE を返します。
?isRelativeObjPositionMaintained	論理値	オブジェクトの相対位置が維持される場合は TRUE、そうでない場合は FALSE を返します。
?isPageHeaderExported	論理値	ページヘッダがエクスポートされる場合は TRUE、そうでない場合は FALSE を返します。
?isPageHeaderSimplified	論理値	ページヘッダが簡略化される場合は TRUE、そうでない場合は FALSE を返します。
?isShowGroupOutlines	論理値	グループのアウトラインが表示される場合は TRUE、そうでない場合は FALSE を返します。

4.5.4 リッチテキスト形式 (RTF) パラメータ

以下のパラメータを使用して、リッチテキスト形式 (RTF) でエクスポートするレポートに関する情報を返すことができます。

オプションパラメータ	型	説明
? isPageBreakAfterEachReportPage	論理値	各レポートページ後にページ区切りがある場合は TRUE、そうでない場合は FALSE を返します。

4.5.5 タブ区切りテキスト (TTX) パラメータ

以下のパラメータを使用して、タブ区切りテキスト (TTX) 形式でエクスポートするレポートを書式設定したり、そのレポートに関する情報を返すことができます。

オプションパラメータ	型	説明
?charactersPerInch=<VALUE>	整数	<VALUE> は、1 インチあたりの文字数を設定します。
?baseAreaType=<VALUE>	整数、文字列	<p><VALUE> は、以下の値の 1 つを使用して、エクスポート文字セットタイプを設定します。</p> <ul style="list-style-type: none"> 0 = UTF-8 1 = UTF=16LE 2 = UTF16BE <p>i 注記 他の値を使用すると、例外がスローされます。</p>
?insertFormFeedCharacter	論理値	フィード間でカスタム区切り文字が使用されている場合は <TRUE>、そうでない場合は <FALSE> を返します。
?minimumLinesPerPage=<VALUE>	整数	<VALUE> は、ページあたりの最小行数を設定する整数です。
?pageAreaExportType=<VALUE>	整数	<p><VALUE> は、以下の値の 1 つを使用して、ページ領域のエクスポートタイプを設定します。</p> <ul style="list-style-type: none"> 0 = AsInReport 1 = OncePerReport 2 = DoNotExport

オプションパラメータ	型	説明
		i 注記 他の値を使用すると、例外がスローされます。

4.5.6 XML パラメータ

以下のパラメータを使用して、RTF 形式でエクスポートするレポートに関する情報を返すことができます。

オプションパラメータ	型	説明
?apply/XSLTIndex=1		0 から始まるインデックスを使用して、エクスポートされた XML に XSL 変換を適用します。
?apply/XSLTName=<VALUE>	文字列	XSLT の名前を使用して、エクスポートされた XML に XSL 変換を適用します。 <VALUE> は、XSLT の名前を表す文字列です。

4.6 インタラクティブパラメータ

レポートにインタラクティブパラメータが含まれる場合は、これらを変更して、エクスポート、行、総計、またはグループ要求から返されるデータをフィルタ処理することができます。これらの要求のインタラクティブパラメータ値を変更するには、?

<PARAMETERNAME>=<VALUES> を使用して、要求 URL にパラメータを追加します。

i 注記

<PARAMETERNAME> はパラメータの名前、<VALUES> はパラメータに設定する値に置き換えてください。

インタラクティブパラメータ値は、レポートの新しいインスタンスを作成するときに変更できます。パラメータ値を設定するには、レポートインスタンス作成フォームを GET し、返されたフォームにパラメータ値を設定します。次に、フォームを POST して、指定したパラメータ値を使用する新しいインスタンスを作成する必要があります。

パラメータでは、次の値がサポートされています。

サポートされる値	例
Null	MyParameter=null
Edm.Boolean	MyParameter=true

サポートされる値	例
Edm.DateTime	MyTimestamp=datetime'2000-12-12T12:34:56.403'
Edm.Double	MyCost=7.89
Edm.String	MyCountry='Canada'
Edm.Time	MyTime=time'PT17H'
日付	MyDate=datetime'2010-02-26T17:08:53'

i 注記

日付値には、対応する OData プリミティブ型がありません。日付値は、Edm.DateTime を使用して表されます。時間部分は有効である必要がありますが、表される日付には影響しません。

範囲パラメータ値は、角かっこを使用して指定します。値はカンマで区切ります。下限値または上限値を範囲から除外する場合は、丸かっこを使用します。

範囲	例
範囲の両端を制限する	MyLunchBreak=[time'PT12H',time'PT13H']
範囲の開始を指定しない	MyMorning=[,time'PT12H']
範囲の終了を指定しない	MyEvening=[time'PT18H',]
範囲の開始値を除外する	MyLunchBreak=(time'PT12H', time'PT13H']
範囲の終了値を除外する	MyLunchBreak=[time'PT12H', time'PT13H')

複数のパラメータ値がある場合は、カンマで区切って並べます。その中の値は、スカラー値、範囲値、または両方の組み合わせを指定できます。

```
MyVacationDays=datetime'2011-03-04T00:00',datetime'2011-03-07T00:00',
[dateTime'2011-03-17T00:00',dateTime'2011-03-21T00:00']
```

例

GET <http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows?<PARAMETERNAME>=<VALUES>>

GET 要求を行って、指定したパラメータでフィルタ処理された行を取得します。

要求:

- メソッド: GET
- URL: http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows?Contact_title='Mr.'
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答:

- 要求に設定したパラメータによってフィルタ処理された行を含むエントリが返されます。

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
      xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices">
```

```

    xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata">
    <title type="text">Rows</title>
    <id>http://localhost:6405/biprws/infostore/7422/rpt/data.svc/Rows?
    Contact_Title='Mr.'</id>
    <link href="Rows?Contact_Title='Mr.'" rel="self" title="Rows"></link>
    <entry>
    <id>http://localhost:6405/biprws/infostore/7422/rpt/data.svc/Rows(0)?
    Contact_Title='Mr.'</id>
    <title type="text">0</title>
    <author><name></name></author>
    <updated>2011-04-26T17:23:06.000</updated>
    <link href="Rows(0)?Contact_Title='Mr.'" rel="self" title="Row"></link>
    <category term="InteractiveKKNNG.Row"
    scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></
    category>
    <content type="application/xml">
    <m:properties>
    <d:ID>0</d:ID>
    <d:Contact_First_Name>Markus</d:Contact_First_Name>
    <d:Contact_Title>Mr.</d:Contact_Title>
    <d:Country>USA</d:Country>
    <d>Last_Year_s_Sales>12013.955</d>Last_Year_s_Sales>
    </m:properties>
    </content>
    </entry>
    </feed>

```

関連情報

[エクスポート \[57 ページ\]](#)

[行 \[75 ページ\]](#)

[レポートインスタンス \[54 ページ\]](#)

[総計 \[68 ページ\]](#)

[グループ \[70 ページ\]](#)

4.7 OData プロトコル

Open Data Protocol (OData) は、レポートデータのクエリと更新に使用されます。OData は、ユーザが任意のプログラミング言語または Web ブラウザを使用してデータの検出、データ間の移動、データの取得を実行できるようにする AtomPub の拡張機能セットです。

4.7.1 OData サービスドキュメントへのアクセス

OData サービスのルートにあるサービスドキュメントを返します。

URI	操作	サポートされる形式	AtomPub か
<baseURI>/infostore/ <reportID>/rpt/ data.svc	GET	XML、JSON	いいえ

例

GET <http://<baseURI>/infostore/<reportID>/rpt/data.svc>

GET 要求を行って、OData サービスドキュメントを取得します。

要求:

- メソッド: GET
- URL: <http://<baseURI>/infostore/<reportID>/rpt/data.svc>
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答:

- OData サービスのルートから提供される情報が格納されたエントリ。

```
<?xml version="1.0" encoding="utf-8"?>
<app:service xmlns:app="http://www.w3.org/2007/app"
  xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom"
  xml:base="http://localhost:6405/biprws/infostore/6311/rpt/data.svc">
  <workspace xmlns="http://www.w3.org/2007/app">
    <atom:title>Default</atom:title>
    <collection href="Rows">
      <atom:title>Rows</atom:title>
    </collection>
    <collection href="GrandTotals">
      <atom:title>GrandTotals</atom:title>
    </collection>
  </workspace>
</app:service>
```

4.7.2 総計

総計コレクションを XML 形式または JSON 形式で返します。総計コレクションには、レポートのヘッダとフッタにあるすべての集計が含まれます。

注記

積算は、このコレクションに含まれません。

URI	操作	サポートされる形式	AtomPub か
<baseURI>/infostore/ <reportID>/rpt/ data.svc/GrandTotals	GET	XML、JSON	はい

オプションパラメータ	型	説明
GrandTotals(0)/<FIELDNAME>	文字列	総計に関連付けられたフィールド値を返します。<FIELDNAME> は、取得するフィールドを指定する文字列です。
GrandTotals(0)/<FIELDNAME>/ \$value	文字列	総計に関連付けられた値のみを返します。<FIELDNAME> は、取得するフィールドを指定する文字列です。



例

GET http://<baseURI>/infostore/<reportID>/rpt/data.svc/GrandTotals

レポート内の総計を取得するための GET 要求を行います。

要求:

- メソッド: GET
- URL: http://<baseURI>/infostore/<reportID>/rpt/data.svc/GrandTotals
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答:

- レポートの総計を含むオブジェクトが返されます。

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
      xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
      xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata">
  <title type="text">GrandTotals</title>
  <id>http://localhost:6405/biprws/infostore/6141/rpt/data.svc/GrandTotals</id>
  <link href="GrandTotals" rel="self" title="GrandTotals"></link>
  <entry>
    <id>http://localhost:6405/biprws/infostore/6141/rpt/data.svc/
    GrandTotals(0)</id>
    <title type="text">0</title>
    <author><name></name></author>
    <updated>2011-01-28T07:58:55.000</updated>
    <link href="GrandTotals(0)" rel="self" title="GrandTotal"></link>
    <category term="SummariesALLSectionsNG_rpt.GrandTotal"
      scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></
    category>
    <content type="application/xml">
      <m:properties>
        <d:ID>0</d:ID>
        <d:Count_Dealer_Price>331</d:Count_Dealer_Price>
        <d:DistinctCount_List_Price>121</d:DistinctCount_List_Price>
      </m:properties>
    </content>
  </entry>
```

```
</feed>
```

4.7.3 グループ

グループパス、グループ名、およびグループコンテンツへのリンクを返します。

i 注記

階層グループは、サポートされていません。

URI	操作	サポートされる形式	AtomPub か
<baseURI>/infostore/ <reportID>/rpt/ data.svc/<GroupName>	GET	XML、JSON	はい

オプション/パラメータ	型	説明
<GroupName> ('<GroupPath>')	整数	<p>指定されたグループに関連付けられたデータを含むリンク付きリストを返します。</p> <p><GroupName> は最上位グループの名前です。<GroupPath> は、取得先のサブグループのインデックスを表す文字列です。グループパスは 0 から始まります。</p> <div><h3>i 注記</h3><p>サブグループを参照するには、URL にサブグループ名とパスを追加します。例:</p><pre>http://<baseURI>/ infostore/ <reportID>/rpt/ data.svc/Country('0')/ Region('0-2')/ City('0-2-4')</pre></div>
<GroupName1> ('<GroupPath>') /\$links/<GroupName2>	整数	<p><GroupName2> で指定されたグループに関連付けられたデータを含むリンク付きリストを返します。<GroupName1> は最上位グループの名前です。<GroupPath> は、最上位グループのインデックスを表す文字列です。グループパスは 0 から始まります。</p>

オプションパラメータ	型	説明
		<p>次の例は、Country('0')に関連付けられたすべての都市のリンク付きリストを返します。</p> <pre>http://<baseURI>/infostore/<reportID>/rpt/data.svc/Country('0')/\$links/City</pre>
<GroupName>? \$select=<PROPERTY>	文字列	<p><PROPERTY> で指定されたプロパティのみを含む行データを返します。次の値を使用できます。</p> <ul style="list-style-type: none"> • * = すべてのプロパティを返します。 • <PROPERTY> = 返されるプロパティの名前です。 • <PROPERTY1>,<PROPERTY2>... = 返されるプロパティの名前です。複数のプロパティがある場合は、それぞれをカンマで区切ります。 <p>i 注記 大文字と小文字は区別されます。</p>
<GroupName>? \$select=<CATEGORY>,<PROPERTY>...	文字列	<p><PROPERTY> で指定されたプロパティのみ、および <CATEGORY> で指定されたカテゴリタイプへのリンクのみを含むグループデータを返します。複数のプロパティがある場合は、それぞれをカンマで区切ります。</p> <p>i 注記 大文字と小文字は区別されます。</p>

4.7.3.1 グループおよびサブグループに移動する

1. 新しい HTTP 要求を作成します。
2. GET メソッドを使用して、最上位グループのコンテンツへのリンクリストを取得するための要求を送信します。
以下はその例です。
 - この例では、Customer_Country が最上位グループです。
 - メソッド: GET
 - URL: <baseURI>/infostore/<reportID>/rpt/data.svc/Customer_Country
 - 要求ヘッダ属性: X-SAP-LogonToken

- 要求本文:なし

応答の例:

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
      xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
      xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata">
  <title type="text">Customer_Country</title>
  <id>http://localhost:6405/biprws/infostore/6246/rpt/data.svc/
Customer_Country</id>
  <link href="Customer_Country" rel="self" title="Customer_Country"></link>
  <entry>
    <id>http://localhost:6405/biprws/infostore/6246/rpt/data.svc/
Customer_Country('0')</id>
    <title type="text">0: Canada</title>
    <author><name></name></author>
    <updated>2010-11-09T18:53:28.000</updated>
    <link href="Customer_Country('0')" rel="self" title="Customer_Country"></
link>
    <category term="ng_groupParams_rpt.Customer_Country"
      scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></
category>
    <link href="Customer_Country('0')/Customer_Region"
      rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
Customer_Region"
      type="application/atom+xml;type=feed" title="Customer_Region">
    </link>
    <link href="Customer_Country('0')/Customer_City"
      rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
Customer_City"
      type="application/atom+xml;type=feed" title="Customer_City">
    </link>
    <content type="application/xml">
      <m:properties>
        <d:ID>0</d:ID>
        <d:Name>Canada</d:Name>
        <d:Sum_of_Amount_of_Delivered_Items_2>377.0</
d:Sum_of_Amount_of_Delivered_Items_2>
      </m:properties>
    </content>
  </entry>
  <entry>
    <id>http://localhost:6405/biprws/infostore/6246/rpt/data.svc/
Customer_Country('1')</id>
    <title type="text">1: England</title>
    <author><name></name></author>
    <updated>2010-11-09T18:53:28.000</updated>
    <link href="Customer_Country('1')" rel="self" title="Customer_Country"></
link>
    <category term="ng_groupParams_rpt.Customer_Country"
      scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></
category>
    <link href="Customer_Country('1')/Customer_Region"
      rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
Customer_Region"
      type="application/atom+xml;type=feed" title="Customer_Region">
    </link>
    <link href="Customer_Country('1')/Customer_City"
      rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
Customer_City"
      type="application/atom+xml;type=feed" title="Customer_City">
    </link>
    <content type="application/xml">
      <m:properties>
        <d:ID>1</d:ID>
        <d:Name>England</d:Name>
```



```

        <d:Sum_of_Amount_of_Delivered_Items_2>1553.0</
d:Sum_of_Amount_of_Delivered_Items_2>
    </m:properties>
</content>
</entry>
</feed>

```

3. GET メソッドを使用して、前のステップで取得したリンクの 1 つに要求を送信します。

要求の例:

- Customer_Country('0') に関連付けられた地域を取得します。
- メソッド: GET
- URL: <baseURI>/infostore/<reportID>/rpt/data.svc/Customer_Country('0')/Customer_Region
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答の例:

```

<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
  xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
  xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata">
  <title type="text">Customer_Region</title>
  <id>http://localhost:6405/biprws/infostore/6246/rpt/data.svc/
Customer_Country('0')/Customer_Region</id>
  <link href="Customer_Country('0')/Customer_Region" rel="self"
title="Customer_Region"></link>
  <entry>
    <id>http://localhost:6405/biprws/infostore/6246/rpt/data.svc/
Customer_Region('0-0')</id>
    <title type="text">0-0: 130</title>
    <author><name></name></author>
    <updated>2010-11-09T18:53:28.000</updated>
    <link href="Customer_Region('0-0') " rel="self" title="Customer_Region"></
link>
    <category term="ng_groupParams_rpt.Customer_Region"
scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></
category>
    <link href="Customer_Region('0-0')/Customer_City"
rel="http://schemas.microsoft.com/ado/2007/08/dataservices/related/
Customer_City"
type="application/atom+xml;type=feed" title="Customer_City">
    </link>
    <content type="application/xml">
      <m:properties>
        <d:ID>0-0</d:ID>
        <d:Name>130</d:Name>
        <d:Sum_of_Amount_of_Delivered_Items_3>377.0</
d:Sum_of_Amount_of_Delivered_Items_3>
      </m:properties>
    </content>
  </entry>
</feed>

```

4. ステップ 2 ~ 3 を繰り返して、グループデータにドリルダウンします。

4.7.4 メタデータ

OData サービスによって公開されるフィード、タイプ、プロパティ、およびコレクションの詳細な説明を含む EDMX ドキュメントを返します。

URI	操作	サポートされる形式	AtomPub か
<baseURI>/infostore/ <reportID>/rpt/ data.svc/\$metadata	GET	XML	はい

例

GET [http://<baseURI>/infostore/<reportID>/rpt/data.svc/\\$metadata](http://<baseURI>/infostore/<reportID>/rpt/data.svc/$metadata)

レポートのメタデータを取得するための GET 要求を行います。

要求:

- メソッド: GET
- URL: [http://<baseURI>/infostore/<reportID>/rpt/data.svc/\\$metadata](http://<baseURI>/infostore/<reportID>/rpt/data.svc/$metadata)
- 要求ヘッダ属性: X-SAP-LogonToken
- 要求本文: なし

応答:

- レポートのメタデータを含む EDMX ドキュメントが返されます。

```
<?xml version="1.0" encoding="utf-8"?>
<edmx:Edmx xmlns:edmx="http://schemas.microsoft.com/ado/2007/06/edmx"
Version="1.0">
  <edmx:DataServices xmlns:m="http://schemas.microsoft.com/ado/2007/08/
dataservices/metadata" m:DataServiceVersion="1.0">
    <Schema xmlns="http://schemas.microsoft.com/ado/2006/04/edm"
xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
xmlns:crds="http://schemas.sap.com/crystalreports/crodataservice"
Namespace="DateTime_rpt">
      <EntityContainer Name="DateTime_rpt_Container"
m:IsDefaultEntityContainer="true">
        <EntitySet Name="Rows" EntityType="DateTime_rpt.Row"
crds:IsGroup="false">
          </EntitySet>
        <EntitySet Name="GrandTotals" EntityType="DateTime_rpt.GrandTotal"
crds:IsGroup="false">
          </EntitySet>
      </EntityContainer>
      <EntityType Name="Row">
        <Key>
          <PropertyRef Name="ID">
            </PropertyRef>
          </Key>
        <Property Name="ID" Type="Edm.Int32">
          </Property>
        <Property Name="Customer_City" Type="Edm.String"
crds:FormulaForm="{SAPBW.Customer¥City}" MaxLength="65534">
          </Property>
      </EntityType>
      <EntityType Name="GrandTotal">
        <Key>
          <PropertyRef Name="ID">
            </PropertyRef>
          </Key>
        <Property Name="ID" Type="Edm.Int32">
          </Property>
      </EntityType>
    </Schema>
  </edmx:DataServices>
```

```
</edmx:Edmx>
```

4.7.5 行

レポートの詳細領域内の行データに対応するデータを返すか更新します。行は、Atom フィードまたは JSON オブジェクトの配列で表されます。

レポートに行を POST する際は、XML と JSON の両方を使用して新しい行情報を指定することもできます。要求本文の形式は、メッセージヘッダの `Content-Type` 属性を使用して指定します。

次の点にご注意ください。

- データベースフィールドデータは、式ではなくレポートにのみ POST できます。
- デリゲートフィールドに POST することはできません。
- 階層グループを含むレポートに POST することはできません。
- OLAP メンバーフィールドに POST することはできません。
- 行データは、レポートの一時インスタンスにのみ POST できます。

i 注記

行 ID は自動的に生成されます。

URI	操作	サポートされる形式	AtomPub か
<code><baseURI>/infostore/ <reportID>/rpt/ data.svc/Rows</code>	GET、POST	XML、JSON	はい

オプション/パラメータ	型	説明
Rows (<INDEX>)	整数	レポートの行を返します。<INDEX> は、フェッチする情報がある行を指定する整数です。
Rows (<INDEX>) /<FIELDNAME>	整数	レポート内のフィールドに関する情報を返します。<INDEX> は、フェッチする情報がある行を指定する整数です。<FIELDNAME> は、返されるフィールド値の名前です。
Rows (<INDEX>) /<FIELDNAME> / \$value	整数	特定の行インデックスのローフィールド値を返します。<INDEX> は、フェッチする情報がある行を指定する整数です。<FIELDNAME> は、返されるフィールド値の名前です。

オプションパラメータ	型	説明
Rows?\$select=<PROPERTY>	文字列	<p><PROPERTY> で指定されたプロパティのみを含む行データを返します。次の値を使用できます。</p> <ul style="list-style-type: none"> • * = すべてのプロパティを返します。 • <PROPERTY> = 返されるプロパティの名前です。 • <PROPERTY1>,<PROPERTY2>,... = 返されるプロパティの名前です。複数のプロパティがある場合は、それぞれをカンマで区切ります。 <p>i 注記 大文字と小文字は区別されます。</p>
<GroupName>?\$select=<CATEGORY>,<PROPERTY>...	文字列	<p><PROPERTY> で指定されたプロパティのみ、および <CATEGORY> で指定されたカテゴリタイプへのリンクのみを含むグループデータを返します。複数のプロパティがある場合は、それぞれをカンマで区切ります。</p> <p>i 注記 大文字と小文字は区別されます。</p>
Rows?\$skip=<VALUE>	整数	指定されたレコード番号から始まる行データチャンクを返します。<VALUE> は、返される最初のレコードを示す整数です。
Rows?\$skiptoken=<VALUE>	整数	指定されたレコード番号から始まる行データチャンクを返します。<VALUE> は、返される最初のレコードを示す整数です。
Rows?\$inlinecount=allpages		すべてのレコードの件数を指定する件数エントリを XML に追加します。
Rows/\$count		レコードの件数を返します。

例

GET [`http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows\(0\)`](http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows(0))

GET メソッドを使用して、レポートの最初の行を要求します。

要求:

- メソッド: GET
- URL: [`http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows\(0\)`](http://<baseURI>/infostore/<reportID>/rpt/data.svc/Rows(0))

- 必須要求ヘッダ属性:X-SAP-LogonToken
- オプションの要求ヘッダ属性:Accept
- 要求本文:なし

応答:

- レポートの 2 行目に関する情報を含むエントリが返されます。

```
<?xml version="1.0" encoding="UTF-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
  xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
  xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata">
  <id>http://localhost:6405/biprws/infostore/6188/rpt/data.svc/Rows(1)</id>
  <title type="text">1</title>
  <author><name></name></author>
  <updated>2010-12-09T20:10:04.000</updated>
  <link href="Rows(0)" rel="self" title="Row"></link>
  <category term="EmployeeData.Row" scheme="http://schemas.microsoft.com/ado/2007/08/dataservices/scheme"></category>
  <content type="application/xml">
    <m:properties>
      <d:ID>1</d:ID>
      <d:RTotal_max_emergency_last_name>Buchanan</
d:RTotal_max_emergency_last_name>
      <d:RTotal_sum_salary>110000.0</d:RTotal_sum_salary>
      <d:Employee_Birth_Date>1960-03-13T00:00:00.000</d:Employee_Birth_Date>
      <d:Employee_Emergency_Contact_Last_Name>Hellstern</
d:Employee_Emergency_Contact_Last_Name>
      <d:Employee_Employee_Id>10.0</d:Employee_Employee_Id>
      <d:Employee_Extension>7559</d:Employee_Extension>
      <d:Employee_First_Name>Albert</d:Employee_First_Name>
      <d:Employee_Last_Name>Hellstern</d:Employee_Last_Name>
      <d:Employee_Position>Business Manager</d:Employee_Position>
      <d:Employee_Reports_To>2.0</d:Employee_Reports_To>
      <d:Employee_Salary>60000.0</d:Employee_Salary>
      <d:Employee_Supervisor_Id>2.0</d:Employee_Supervisor_Id>
      <d:Employee_Home_Phone>(206) 555-4869</d:Employee_Home_Phone>
    </m:properties>
  </content>
</entry>
```

例

POST <http://<baseURI>/infostore/<reportID>/rpt/<instanceID>/data.svc/Rows>

レポートのインスタンスに行を POST します。

要求:

- メソッド:POST
- URL:<http://<baseURI>/<reportID>/rpt/<instanceID>/data.svc/Rows>
- 要求ヘッダ属性:

```
accept
accept-charset: UTF-8
content-type: application/atom+xml
host: <HostName>
content-length: 1000
X-SAP-LogonToken: COMMANDCOM-LCM:6400@{3&2=5328,U3&p=40676
.8926203819,Y7&4F=12,U3&63=secEnterprise,0P&66=60,03&68=secEnterprise:
Administrator,0P&qe=100,U3&vz=IVD21LbMCB0eRiI4atz9sNL18Ux5anRBdYB9fFv5NrY,UP}
```

- 要求本文: 行情報を追加します。

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?>
<entry xmlns:d="http://schemas.microsoft.com/ado/2007/08/dataservices"
xmlns:m="http://schemas.microsoft.com/ado/2007/08/dataservices/metadata"
xmlns="http://www.w3.org/2005/Atom">
<content type="application/xml">
<m:properties>
<d:Orders_Order_Date m:type="Edm.DateTime">2011-01-28T00:00:00.000</
d:Orders_Order_Date>
</m:properties>
</content>
</entry>
```

i 注記

term="Xtreme_OrderDate.Row" は、レポート内の行エンティティの名前です。レポートメタデータを取得して、レポート内のエンティティの名前を検索できます。

応答:

- レポートに追加した行を含むエントリが返されます。

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns="http://www.w3.org/2005/Atom" xmlns:d="http://
schemas.microsoft.com/ado/2007/08/dataservices" xmlns:m="http://
schemas.microsoft.com/ado/2007/08/dataservices/metadata">
  <id>http://localhost:6405/biprws/1667/rpt/data.svc/Rows(523)</id>
  <title type="text">523</title>
  <category term="CRoDataService.Row" scheme="http://schemas.microsoft.com/ado/
2007/08/dataservices/scheme">
  </category>
  <content type="application/xml">
    <m:properties>
      <d:ID>523</d:ID>
      <d:Orders_Order_Date>2011-01-28T00:00:00.000</d:Orders_Order_Date>
    </m:properties>
  </content>
</entry>
```

重要免責事項および法的情報

コードサンプル

この文書に含まれるソフトウェアコード及び / 又はコードライン / 文字列 (「コード」) はすべてサンプルとしてのみ提供されるものであり、本稼動システム環境で使用することが目的ではありません。「コード」は、特定のコードの構文及び表現規則を分かりやすく説明及び視覚化することのみを目的としています。SAP は、この文書に記載される「コード」の正確性及び完全性の保証を行いません。更に、SAP は、「コード」の使用により発生したエラー又は損害が SAP の故意又は重大な過失が原因で発生させたものでない限り、そのエラー又は損害に対して一切責任を負いません。

アクセシビリティ

この SAP 文書に含まれる情報は、公開日現在のアクセシビリティ基準に関する SAP の最新の見解を表明するものであり、ソフトウェア製品のアクセシビリティ機能の確実な提供方法に関する拘束力のあるガイドラインとして意図されるものではありません。SAP は、この文書に関する一切の責任を明確に放棄するものです。ただし、この免責事項は、SAP の意図的な違法行為または重大な過失による場合は、適用されません。さらに、この文書により SAP の直接的または間接的な契約上の義務が発生することは一切ありません。

ジェンダーニュートラルな表現

SAP 文書では、可能な限りジェンダーニュートラルな表現を使用しています。文脈により、文書の読者は「あなた」と直接的な呼ばれ方をされたり、ジェンダーニュートラルな名詞 (例: 「販売員」又は「勤務日数」) で表現されます。ただし、男女両方を指すとき、三人称単数形の使用が避けられない又はジェンダーニュートラルな名詞が存在しない場合、SAP はその名詞又は代名詞の男性形を使用する権利を有します。これは、文書を分かりやすくするためです。

インターネットハイパーリンク

SAP 文書にはインターネットへのハイパーリンクが含まれる場合があります。これらのハイパーリンクは、関連情報を見いだすヒントを提供することが目的です。SAP は、この関連情報の可用性や正確性又はこの情報が特定の目的に役立つことの保証は行いません。SAP は、関連情報の使用により発生した損害が、SAP の重大な過失又は意図的な違法行為が原因で発生したものでない限り、その損害に対して一切責任を負いません。すべてのリンクは、透明性を目的に分類されています (<http://help.sap.com/disclaimer> を参照)。

[go.sap.com/registration/
contact.html](http://go.sap.com/registration/contact.html)

© 2016 SAP SE or an SAP affiliate company. All rights reserved.

本書のいかなる部分も、SAP SE 又は SAP の関連会社の明示的な許可なくして、いかなる形式でも、いかなる目的にも複製又は伝送することはできません。本書に記載された情報は、予告なしに変更されることがあります。SAP SE 及びその頒布業者によって販売される一部のソフトウェア製品には、他のソフトウェアベンダーの専有ソフトウェアコンポーネントが含まれています。製品仕様は、国ごとに変わる場合があります。

これらの文書は、いかなる種類の表明又は保証もなしで、情報提供のみを目的として、SAP SE 又はその関連会社によって提供され、SAP 又はその関連会社は、これら文書に関する誤記脱落等の過失に対する責任を負うものではありません。SAP 又はその関連会社の製品及びサービスに対する唯一の保証は、当該製品及びサービスに伴う明示的保証がある場合に、これに規定されたものに限られます。本書のいかなる記述も、追加の保証となるものではありません。

本書に記載される SAP 及びその他の SAP の製品やサービス、並びにそれらの個々のロゴは、ドイツ及びその他の国における SAP SE (又は SAP の関連会社) の商標若しくは登録商標です。本書に記載されたその他すべての製品およびサービス名は、それぞれの企業の商標です。

商標に関する情報および表示の詳細については、<http://www.sap.com/corporate-en/legal/copyright/index.epx> をご覧ください。