



InterSystems IRIS Business Intelligence のクライアント側 API

Version 2024.1
2024-06-03

InterSystems IRIS Business Intelligence のクライアント側 API
InterSystems IRIS Data Platform Version 2024.1 2024-06-03
Copyright © 2024 InterSystems Corporation
All rights reserved.

InterSystems®, HealthShare Care Community®, HealthShare Unified Care Record®, IntegratedML®, InterSystems Caché®, InterSystems Ensemble®, InterSystems HealthShare®, InterSystems IRIS®, および TrakCare は、InterSystems Corporation の登録商標です。HealthShare® CMS Solution Pack™ HealthShare® Health Connect Cloud™, InterSystems IRIS for Health™, InterSystems Supply Chain Orchestrator™, および InterSystems TotalView™ For Asset Management は、InterSystems Corporation の商標です。TrakCare は、オーストラリアおよび EU における登録商標です。

ここで使われている他の全てのブランドまたは製品名は、各社および各組織の商標または登録商標です。

このドキュメントは、インターシステムズ社(住所: One Memorial Drive, Cambridge, MA 02142)あるいはその子会社が所有する企業秘密および秘密情報を含んでおり、インターシステムズ社の製品を稼動および維持するためにのみ提供される。この発行物のいかなる部分も他の目的のために使用してはならない。また、インターシステムズ社の書面による事前の同意がない限り、本発行物を、いかなる形式、いかなる手段で、その全てまたは一部を、再発行、複製、開示、送付、検索可能なシステムへの保存、あるいは人またはコンピュータ言語への翻訳はしてはならない。

かかるプログラムと関連ドキュメントについて書かれているインターシステムズ社の標準ライセンス契約に記載されている範囲を除き、ここに記載された本ドキュメントとソフトウェアプログラムの複製、使用、廃棄は禁じられている。インターシステムズ社は、ソフトウェアライセンス契約に記載されている事項以外にかかるソフトウェアプログラムに関する説明と保証をするものではない。さらに、かかるソフトウェアに関する、あるいはかかるソフトウェアの使用から起こるいかなる損失、損害に対するインターシステムズ社の責任は、ソフトウェアライセンス契約にある事項に制限される。

前述は、そのコンピュータソフトウェアの使用およびそれによって起こるインターシステムズ社の責任の範囲、制限に関する一般的な概略である。完全な参照情報は、インターシステムズ社の標準ライセンス契約に記載され、そのコピーは要望によって入手することができる。

インターシステムズ社は、本ドキュメントにある誤りに対する責任を放棄する。また、インターシステムズ社は、独自の裁量にて事前通知なしに、本ドキュメントに記載された製品および実行に対する代替と変更を行う権利を有する。

インターシステムズ社の製品に関するサポートやご質問は、以下にお問い合わせください:

InterSystems Worldwide Response Center (WRC)
Tel: +1-617-621-0700
Tel: +44 (0) 844 854 2917
Email: support@InterSystems.com

目次

1 Business Intelligence の使用法	1
1.1 Web アプリケーションの作成	1
1.2 Business Intelligence JavaScript API の概要	1
1.2.1 Business Intelligence 接続の作成	2
1.2.2 Business Intelligence データ・コントローラの作成および使用	2
1.3 Business Intelligence REST API の概要	4
1.3.1 キューブ名と KPI 名でのスラッシュの使用	5
1.3.2 応答オブジェクトに関する注意事項	5
DeepSee.js	7
DeepSeeConnector	8
DeepSeeDataController	9
DeepSeeResultSet	12
DeepSeeUtils	15
Business Intelligence REST API	19
GET /Data/GetDSTIME	20
GET /Data/TestConnection	21
GET /Info/TestConnection	22
POST /Command/BuildAllRegisteredGroups	23
POST /Command/BuildCube	25
POST /Command/BuildOneRegisteredGroup	26
POST /Command/BuildRegistryMap	27
POST /Command/GetActiveRegistry	29
POST /Command/GetCubeSize	30
POST /Command/GetLastUpdate	31
POST /Command/getSynchScheduleParameters	32
POST /Command/IsValidCubeSchedule	33
POST /Command/IsValidGroup	34
POST /Command/RepairBuild	38
POST /Command/SetActiveRegistry	39
POST /Command/SynchronizeCube	40
POST /Command/WriteToRegistry	41
POST /Data/KPIExecute	42
POST /Data/MDXCancelQuery/:queryID	44
POST /Data/MDXDrillthrough	45
POST /Data/MDXExecute	47
POST /Data/MDXUpdateResults/:queryID	50
POST /Data/PivotExecute	51
POST /Info/Cubes	52
POST /Info/Dashboards	54
POST /Info/FilterMembers/:datasource/:filterSpec	55
POST /Info/Filters/:datasource	57
POST /Info/KPIs	58
POST /Info/ListingFields/:cube	59
POST /Info/Listings/:cube	60
POST /Info/Measures/:cube	62
POST /Info/NamedFilters/:cube	63

POST /Info/Pivots	64
POST /Info/PivotVariableDetails/:cube/:variable	65
POST /Info/PivotVariables/:cube	66
POST /Info/QualityMeasures/:cube	67

1

Business Intelligence の使用法

ここでは、InterSystems IRIS® データ・プラットフォームの [Business Intelligence](#) 用の JavaScript API と REST API の概要を示します。これらの API を使用すると、[MDX クエリ](#)を実行して、Business Intelligence [モデルの要素](#)に関する情報を取得できます。

1.1 Web アプリケーションの作成

どのようなシナリオでも (JavaScript API を使用するのか REST サービスを直接使用するのかにかかわらず)、Web アプリケーションは要求を処理する必要があります。システム定義の Web アプリケーション (`/api/deepsee`) を使用することも、それとは別の Web アプリケーションを作成して使用することもできます。この Web アプリケーションの要件は以下のとおりです。

- ・ クライアントのファイルは、この Web アプリケーションによって提供されるディレクトリ構造内に配置する必要があります。
- ・ **[ディスパッチ・クラス]** オプションを指定する必要があります。このオプションでは、この Web アプリケーションが REST 要求を処理する方法を指定します。Business Intelligence の REST 要求については、以下のいずれかを使用します。
 - **%Api.DeepSee** - 目的のクライアント・アプリケーションが異なるネームスペースに接続できる必要がある場合は、このクラスを使用します。この場合は、InterSystems IRIS サーバに接続する際に、使用するネームスペースを指定する必要があります。

システム定義の Web アプリケーション (`/api/deepsee`) は、このディスパッチ・クラスを使用します。

- **%DeepSee.REST.v1** - REST 要求が特定のネームスペース (この Web アプリケーションのネームスペース) に結び付けられる必要がある場合は、このクラスを使用します。

1.2 Business Intelligence JavaScript API の概要

Business Intelligence JavaScript API は、`install-dir/CSP/broker` ディレクトリにある **DeepSee.js** ファイルによって提供されます。この JavaScript ライブラリを使用すると、JavaScript ベースのクライアントから Business Intelligence を操作できます。このライブラリ内の関数は、Business Intelligence 用の REST ベース API のラップです ([REST API](#) を直接使用することもできます)。

このライブラリを使用する手順は以下のとおりです。

1. Web アプリケーションを作成します (前のセクションを参照)。
または、インストール内容に含まれている `/api/deepsee` という Web アプリケーションを使用します。
2. JavaScript クライアント・コード内で、以下の手順を実行します。
 - a. **DeepSee.js** ファイルと **zenCSLM.js** ファイルをインクルードします
 - b. Business Intelligence 接続オブジェクトを作成します。これには、InterSystems IRIS サーバに接続するために必要な情報が含まれています。
 - c. この接続オブジェクトを使用する Business Intelligence データ・コントローラ・オブジェクトを作成します。
このデータ・コントローラ・オブジェクトを使用すると、Business Intelligence データ・ソースを操作できます。このデータ・ソースは、MDX クエリまたはピボット・テーブル名を通じて指定します。
 - d. データ・コントローラの `runQuery()` メソッドを使用します。データ・ソースが MDX クエリである場合、Business Intelligence はそのクエリを実行します。データ・ソースがピボット・テーブルである場合、Business Intelligence はそのピボット・テーブルで定義されたクエリを実行します。
 - e. データ・コントローラ・オブジェクトの他のメソッドを実行して、クエリ結果を調べたり、ドリル・ダウンやドリル・スルーなどを実行します。

次のサブセクションで詳細を説明します。

DeepSee.js ライブラリは、Business Intelligence のモデル要素に関する情報を提供するユーティリティ関数も提供します。これらを使用して、使用可能なキューブのリストや、キューブ内の使用可能なメジャーのリストなどを取得します。

1.2.1 Business Intelligence 接続の作成

Business Intelligence 接続オブジェクトを作成するには、次のようなコードを使用します。

```
connection = new DeepSeeConnection(username,password,host,application,namespace);
```

以下はその説明です。

- ・ `username` は、指定されたホストにアクセスできる InterSystems IRIS ユーザ名です。
- ・ `password` は、関連付けられたパスワードです。
- ・ `host` は、InterSystems IRIS を実行しているマシンのサーバ名です。
- ・ `application` は、Web アプリケーションの名前です。
- ・ `namespace` は、アクセスするネームスペースの名前です。Web アプリケーションがネームスペースに結び付けられている場合は、この引数は不要です。

1.2.2 Business Intelligence データ・コントローラの作成および使用

データ・コントローラ・オブジェクトを使用すると、Business Intelligence データ・ソースを操作できます。主な操作内容は以下のとおりです。

- ・ ページ・ロジックの適切な部分で (ページの読み込み時やボタンの押下時など)、Business Intelligence データ・コントローラを作成して、クエリを実行します。
データ・コントローラを作成する際は、データが得られたときに実行される 1 つまたは 2 つのコールバック関数を指定します。`finalCallback` は必須ですが、`pendingCallback` はオプションです。
- ・ 保留中の結果が得られる場合、Business Intelligence は `pendingCallback` で指定されたメソッドを呼び出します (指定されている場合)。

このメソッドは、ユーザによって記述され、データ・コントローラ・オブジェクトに含まれている結果を使用します。このメソッドは通常、ページ・コンテンツを描画します。

- ・ クエリが完了したら、Business Intelligence は `finalCallback` で指定されたメソッドを呼び出します。

このメソッドは、ユーザによって記述され、データ・コントローラ・オブジェクトに含まれている結果を使用します。このメソッドは通常、ページ・コンテンツを描画します。

クエリを実行するメソッドはすべて、上記の仕組みを使用します。詳細と例については、サブセクションを参照してください。他のメソッドは同期的にデータを返します。

1.2.2.1 Business Intelligence データ・コントローラの作成とクエリの実行

クライアント・コードの適切な部分で (ページ初期化ロジック内など)、以下の手順を実行します。

1. 以下のプロパティを持つ構成オブジェクトを作成します。
 - ・ `connection` – Business Intelligence データ・コネクタ・オブジェクトの名前を指定します。[前のセクション](#)を参照してください。
 - ・ `widget` – データ・コントローラを使用するページ上の HTML 要素の ID を指定します。
 - ・ `type` – データ・ソースのタイプとして 'MDX' または 'PIVOT' を指定します。
 - ・ `initialMDX` – MDX SELECT クエリを指定します (このプロパティは `type` が 'MDX' の場合に使用します)。
 - ・ `pivotName` – ピボット・テーブルの論理名を指定します (このプロパティは `type` が 'PIVOT' の場合に使用します)。
 - ・ `showTotals` – 合計を表示するかどうかを指定します。`true` または `false` を指定します。
2. 次のようなコードを使用してデータ・コントローラ・オブジェクトを作成します。

```
var dc = new DeepSeeDataController(configuration,finalCallback,pendingCallback);
```

ここで、`configuration` は前の手順で作成した構成オブジェクトであり、`finalCallback` はこのページ上のコールバック関数の名前であり、`pendingCallback` はこのページ上のもう 1 つのコールバック関数の名前です。`finalCallback` は必須ですが、`pendingCallback` はオプションです。

3. データ・コントローラの `runQuery()` メソッドを呼び出します。または、クエリを実行する他のメソッド (`runDrillDown()` や `runListing()` など) を実行します。

次に例を示します。

```
function initializePage() {
    ...
    configuration.connection = new
DeepSeeConnection(username,password,host,application,namespace);
    dc = new DeepSeeDataController(configuration,drawChart);
    dc.runQuery();
}
```

1.2.2.2 データ・コントローラによって返されたデータの使用

ページでは、前の手順で示したコールバック関数を実装する必要もあります。これらのコールバックは、データ・コントローラ・オブジェクトから取得されたデータを使用して、ページを必要に応じて更新します。

それぞれの場合に、[データ・コントローラ・オブジェクト](#)が引数として関数に渡されます。

以下は、部分的な例です。

```
function drawChart(dataController) {
    var resultSet = dataController.getCurrentData();
    ...
    var chartDataPoint;
    var chartLabel;
    var chartData = [];

    for (var i = 1; i <= resultSet.getRowCount(); ++i) {
        for (var j = 1; j <= resultSet.getColumnCount(); ++j) {
            chartDataPoint = resultSet.getOrdinalValue(i,j);
            chartLabel = resultSet.getOrdinalLabel(2,i);
            chartData[chartData.length] = { "country":chartLabel[0], "revenue":chartDataPoint };
        }
    }
    ...
}
```

データ・コントローラの `getCurrentData()` メソッドは、もう 1 つのオブジェクトである結果セット・オブジェクトを返します。このオブジェクトは、クエリの結果を調べるためのメソッドを提供します。上記の例では、これらのメソッドのいくつかを示しています。

1.3 Business Intelligence REST API の概要

内部的には、[JavaScript API](#) は Business Intelligence REST API を使用します。この REST API は、以下のように直接使用することもできます。

1. [Web アプリケーション](#)を作成します。

または、インストール内容に含まれている `/api/deepsee` という Web アプリケーションを使用します。

2. JavaScript クライアント・コード内で、HTTP 要求を作成して、目的のターゲット REST サービスに送信します。

ディスパッチ・クラス `%Api.DeepSee` を使用している場合は、次の形式のターゲット URL を使用します。

```
https://baseUrl/api/deepsee/v1/namespace/RESTcallname
```

ここで、`baseUrl` はインスタンスを参照し、`namespace` はターゲット・ネームスペースであり、`RESTcallname` は実際の REST 呼び出し (`/Info/Cubes` など) です。次に例を示します。

```
/mycompany/api/deepsee/v1/myapplication/Info/Cubes
```

ディスパッチ・クラス `%DeepSee.REST.v1` を使用している場合は、次の形式のターゲット URL を使用します。

```
/baseUrl/api/deepsee/v1/RESTcallname
```

次に例を示します。

```
/mycompany/api/deepsee/v1/Info/Cubes
```

注釈 クライアントは、JSON を受け入れる必要があります。要求の `Accept` ヘッダは、`application/json` を指定するか、形式を宣言しない必要があります。

3. 応答オブジェクトを調べて、適宜使用します。

1.3.1 キューブ名と KPI 名でのスラッシュの使用

キューブなどのアイテムの論理名では、スラッシュ (/) が比較的によく使用されます。スラッシュ文字は、フォルダ名と短いアイテム名を区切るための記号であるからです。例えば、キューブの論理名として `RelatedCubes/Patients` が考えられます。

URL パラメータ(および要求の本文)では、これらの論理名を変更せずに直接使用できます。該当する Business Intelligence REST サービスでは、スラッシュが含まれた論理名が考慮に入れられます。ただし、ロジックでは名前付け規約(使用する予定の REST サービスに応じて異なります)に従うことが要求されます。具体的には、別の論理名で使用されているフォルダ名と同じ論理名を持つアイテムは避けてください。例えば、`mycubes/test/test1` という名前のアイテムがある場合は、`mycubes/test` という名前のアイテムを避ける必要があります。

この制約の理由は、論理名の後ろに別の引数を使用している REST サービスを使用する際に、その名前の最初の部分が既存のアイテム名と一致する場合は、その名前の一部は別の引数として解釈されるからです。例えば、次の REST 呼び出しを考えてみてください。

```
https://localhost/api/deepsee/v1/Info/FilterMembers/:mycubename/:filterspec
```

ここで、`mycubename` はキューブの論理名であり、`filterspec` はそのキューブによって提供されるフィルタの仕様です。次に、`mycubes/test/test1` をキューブ名として指定した次の REST 呼び出しを考えてみてください。

```
https://localhost/api/deepsee/v1/Info/FilterMembers/:mycubes/test/test1/:filterspec
```

スラッシュ文字を解釈するために、システムは、最初に `mycubes` という名前のキューブを探してから、`mycubes/test` という名前のキューブを探すというように、順番にキューブを探していきます。システムがこの見かけ上の名前に一致する最初のアイテムを見つければ、REST 呼び出しではそのアイテムが使用され、文字列の残り部分は次の引数として解釈されます。

1.3.2 応答オブジェクトに関する注意事項

ほとんどの REST 呼び出しについて、応答オブジェクトには `Info` プロパティが含まれています。このプロパティには要求と応答に関する情報が含まれています。このオブジェクトには、次のいずれかに該当する `Error` プロパティが含まれています。

- ・ `NULL` – これはエラーが発生しなかったことを意味します。
- ・ `ErrorCode` プロパティと `ErrorMessage` プロパティが含まれたオブジェクト – このオブジェクトに含まれているエラー情報に基づいて、ユーザは処理を進めるのかどうかと処理の進め方を決定できます。

エラーが発生していない場合は、応答オブジェクトには、要求された値が格納されたオブジェクトである `Result` プロパティも含まれています。

一般に、クライアント・コードではまず `Info.Error` プロパティを確認してから、処理の進め方を決定する必要があります。

例えば、以下のような応答オブジェクトが考えられます(読みやすいように空白を追加しています)。

```
{ "Info":
  { "Error":
    { "ErrorCode": "5001", "ErrorMessage": "ERROR #5001: Cannot find Subject Area: 'SampleCube'" }
  }
}
```

対照的に、エラーが発生していない場合は、`Info.Error` プロパティは `NULL` であり、`Result` には要求した結果が含まれています。次に例を示します。

```
{ "Info":
  { "Error": "",
    "BaseCube": "DemoMDX",
    "SkipCalculated": 0 },
  "Result":
    { "Measures":
      [
        { "name": "%COUNT", "caption": "%COUNT", "type": "integer", "hidden": 0, "factName": "" },
        { "name": "Age", "caption": "Age", "type": "integer", "hidden": 0, "factName": "MxAge" }
      ]
    }
}
```

DeepSee.js

このリファレンス・セクションでは、InterSystems IRIS Business Intelligence の JavaScript API に関する情報を提供します。この API は、**DeepSee.js** ファイルによって提供されます。

DeepSeeConnector

Business Intelligence データ・ソースに接続することを可能にします。

このオブジェクトを使用できる場所

このオブジェクトは、クライアント側の JavaScript コード内で使用できます（そのコードで **DeepSee.js** がインクルードされている場合）。"[DeepSee JavaScript API の概要](#)" を参照してください。

このオブジェクトの作成

Business Intelligence データ・コネクタ・オブジェクトを作成するには、以下のようなコードを使用します。

```
var connection = new DeepSeeConnection(username,password,host,application,namespace);
```

以下はその説明です。

- ・ username は、指定されたホストにアクセスできる InterSystems IRIS® データ・プラットフォーム・ユーザ名です。
- ・ password は、関連付けられたパスワードです。
- ・ host は、InterSystems IRIS を実行しているマシンのサーバ名です。
- ・ application は、Web アプリケーションの名前です。
- ・ namespace は、アクセスするネームスペースの名前です。

このオブジェクトのプロパティ

このオブジェクトには、以下のプロパティがあります。

- ・ username は、指定されたホストにアクセスできる InterSystems IRIS ユーザ名です。
- ・ password は、関連付けられたパスワードです。
- ・ path は、Web サービスのベース URL です。

このオブジェクトのメソッド

データ・コネクタ・オブジェクトにはメソッドはありません。

DeepSeeDataController

Business Intelligence データ・ソースを操作することを可能にします。

このオブジェクトを使用できる場所

このオブジェクトは、クライアント側の JavaScript コード内で使用できます（そのコードで **DeepSee.js** がインクルードされている場合）。"[DeepSee JavaScript API の概要](#)" を参照してください。

このオブジェクトの作成

Business Intelligence データ・コントローラ・オブジェクトを作成するには、次のようなコードを使用します。

```
var controller = new DeepSeeDataController(configuration,finalCallback,pendingCallback);
```

以下はその説明です。

- ・ configuration は、以下のプロパティを持つオブジェクトです。
 - connection - Business Intelligence [データ・コネクタ・オブジェクト](#)。
 - widget - データ・コントローラを使用するページ上の HTML 要素の ID を指定します。
 - type - データ・ソースのタイプとして 'MDX' または 'PIVOT' を指定します（大文字と小文字が区別されます）。
 - initialMDX - MDX SELECT クエリを指定します（このプロパティは type が 'MDX' の場合に指定します）。
 - pivotName - ピボット・テーブルの論理名を指定します（このプロパティは type が 'PIVOT' の場合に指定します）。
 - showTotals - 合計を表示するかどうかを指定します。このプロパティには true または false を指定します。
- ・ finalCallback は、このページ上のコールバック関数の名前です。
このオブジェクトの runQuery() などのメソッドについては、クエリが完了すると、システムはこの関数を呼び出します。
- ・ pendingCallback (オプション) は、このページ上のもう 1 つのコールバック関数の名前です。
このオブジェクトの runQuery() などのメソッドについては、保留中の結果が得られたときに、システムはこの関数を呼び出します（この引数が指定されている場合）。

このオブジェクトのメソッド

Business Intelligence データ・コントローラ・オブジェクトには、以下の JavaScript メソッドがあります。

applyFilter()

```
applyFilter(filterInfo)
```

ここで、filterInfo は filterName プロパティと filterSpec プロパティが含まれたオブジェクトです。

このメソッドは、データ・コントローラで使用されているフィルタに指定フィルタを追加して、クエリを再実行してから、データ・コントローラ・オブジェクトに関連付けられている[コールバック関数](#)を呼び出します。

このメソッドには戻り値はありません。

attachTotals()

```
attachTotals(rowTotals,columnTotals,reattach)
```

以下はその説明です。

- ・ `rowTotals` には、行の合計をアタッチするかどうかに応じて `true` または `false` を指定します。
- ・ `columnTotals` には、列の合計をアタッチするかどうかに応じて `true` または `false` を指定します。
- ・ `reattach` には、最後に保存された状態を更新するかどうかに応じて `true` または `false` を指定します。
`reattach` が `true` の場合、システムは最後に保存された状態を更新します。`reattach` が `false` の場合、システムは新しい状態オブジェクトをスタックに追加します。

このメソッドは、データ・コントローラ・オブジェクトに合計をアタッチしてから、データ・コントローラ・オブジェクトに関連付けられている[コールバック関数](#)を呼び出します。

このメソッドには戻り値はありません。

`getCurrentData()`

```
getCurrentData()
```

[Business Intelligence 結果セット・オブジェクト](#)を返します。このオブジェクトには、データ・コントローラ・オブジェクトによって現在定義されているクエリの結果が含まれています。[DeepSeeResultSet](#) オブジェクトのリファレンス情報を参照してください。

`getCurrentQueryText()`

```
getCurrentQueryText()
```

データ・コントローラ・オブジェクトで現在定義されているクエリのテキストを返します。

`runDrillDown()`

```
runDrillDown(axis, position)
```

以下はその説明です。

- ・ `axis` は、ドリルダウン操作を実行する軸の番号です。列の場合は 1 を指定し、行の場合は 2 を指定します。
- ・ `position` は、指定された軸上のドリルダウン操作を実行する位置 (先頭は 1) です。

このメソッドは、指定されたドリルダウン操作を実行してから、データ・コントローラ・オブジェクトに関連付けられている[コールバック関数](#)を呼び出します。

このメソッドには戻り値はありません。

`runListing()`

```
runListing(startRow, startCol, endRow, endCol, listingName)
```

以下はその説明です。

- ・ `startRow` と `startCol` は、詳細リストを希望する結果の最初の行と列の番号です。先頭の行または列の場合は 1 を指定します。
- ・ `endRow` と `endCol` は、詳細リストを希望する最後の行と列の番号です。
- ・ `listingName` は、詳細リストの論理名です。

このメソッドは、結果の 1 つ以上のセルについて指定された詳細リストを実行してから、データ・コントローラ・オブジェクトに関連付けられている[コールバック関数](#)を呼び出します。

このメソッドには戻り値はありません。

runQuery()

```
runQuery()
```

データ・コントローラ・オブジェクトで現在定義されているクエリを実行します。このクエリが保留中のときや完了したときに、システムはデータ・コントローラ・オブジェクトに関連付けられている[コールバック関数](#)を呼び出します。

このメソッドには戻り値はありません。

sortResults()

```
sortResults(axis, position, direction, sortType)
```

以下はその説明です。

- ・ axis は、ソートする軸の番号です。列の場合は 1 を指定し、行の場合は 2 を指定します。
- ・ position は、ソートする当該軸上の位置 (先頭は 1) です。
例えば、3 列目を基準にしてソートするには、axis に 1 を指定し、position に 3 を指定します。
- ・ direction は、ソートする方向です。昇順でソートするには 1 を指定し、降順でソートするには -1 を指定します。
- ・ sortType は、ソート方法を指定します。この値が '' または 'numeric' (大文字と小文字は区別されません) である場合は、数値ソートが実行されます。それ以外の場合は、文字列ソートが実行されます。

このメソッドは、指定されたとおりに結果をソートしてから、データ・コントローラ・オブジェクトに関連付けられている[コールバック関数](#)を呼び出します。

このメソッドには戻り値はありません。

undoLastAction()

```
undoLastAction()
```

最後に加えられた変更を元に戻してから、データ・コントローラ・オブジェクトに関連付けられている[コールバック関数](#)を呼び出します。

このメソッドには戻り値はありません。

DeepSeeResultSet

Business Intelligence クエリの結果を操作することを可能にします。

このオブジェクトを使用できる場所

このオブジェクトは、クライアント側の JavaScript コード内で使用できます（そのコードで **DeepSee.js** がインクルードされている場合）。[DeepSee JavaScript API の概要](#)を参照してください。

このオブジェクトの作成

結果セットを作成するには、[データ・コントローラ・オブジェクト](#)の `getCurrentData()` メソッドを呼び出します。

または、次のようなコードを使用します。

```
var resultset = new DeepSeeResultSet(connection,widget,wait,timeout);
```

以下はその説明です。

- ・ `connection` – Business Intelligence [データ・コネクタ・オブジェクト](#)。
- ・ `widget` – [データ・コントローラ](#)を使用するページ上の HTML 要素の ID を指定します。
- ・ `wait` – ブーリアン値。このパラメータの値が `false` の場合は、サーバは直ちに応答します。このパラメータの値が `true` の場合は、クエリが完了するかタイムアウトになるまで、サーバは応答しません。
- ・ `timeout` – 最終の未完了クエリ結果を返すまでサーバが待機する時間を指定します（秒単位）。

このオブジェクトのプロパティ

Business Intelligence の結果セット・オブジェクトには、以下のプロパティがあります。

- ・ `connection` – Business Intelligence [データ・コネクタ・オブジェクト](#)。
- ・ `widget` – [データ・コントローラ](#)を使用するページ上の HTML 要素の ID を指定します。
- ・ `wait` – ブーリアン値。このプロパティの値が `false` の場合は、サーバは直ちに応答します。このプロパティの値が `true` の場合は、クエリが完了するかタイムアウトになるまで、サーバは応答しません。
- ・ `timeout` – 最終の未完了クエリ結果を返すまでサーバが待機する時間を指定します（秒単位）。
- ・ `data` – クエリ結果が含まれています。
- ・ `pollInterval` – 保留中の結果の更新をサーバに要求するまでの待機時間を指定します（ミリ秒単位）。既定は 1000 です。

このオブジェクトのメソッド

Business Intelligence 結果セット・オブジェクトには、以下の JavaScript メソッドがあります。

getColumnCount()

```
getColumnCount()
```

結果セット内の列数を返します。

getCubeName()

```
getCubeName()
```


現在使用されているキューブの名前を返します。

getErrorMessage()

```
getErrorMessage()
```

このメソッドは、応答オブジェクトに含まれているエラー・メッセージ (ある場合) のテキストを返します。

getOrdinalLabel()

```
getOrdinalLabel(axis, position)
```

以下はその説明です。

- ・ axis は、ラベルを取得する軸の番号です。列の場合は 1 を指定し、行の場合は 2 を指定します。
- ・ position は、ラベルを取得する当該軸上の位置です。軸上の先頭位置の場合は 1 を指定します。

指定された軸上の指定された位置のラベルに対応する、文字列の配列を返します。

getOrdinalValue()

```
getOrdinalValue(rowNo,colNo,formatted)
```

以下はその説明です。

- ・ rowNo は行番号です (先頭は 1)。
- ・ colNo は列番号です (先頭は 1)。

指定されたセルの値を返します。

getQueryStatus()

```
getQueryStatus()
```

クエリが完了したかどうかを示す数値を返します。クエリが完了している場合は、この値は 100 です。そうでない場合、この値は 100 未満になります。

getRowCount()

```
getRowCount()
```

結果セット内の行数を返します。

isError()

```
isError()
```

エラーが発生したことが応答オブジェクトによって示されている場合は、このメソッドは true を返します。そうでない場合は、false を返します。

runMDXQuery()

```
runMDXQuery(mdx,finalCallback,pendingCallback,filters)
```

以下はその説明です。

- ・ mdx は MDX SELECT クエリです。

- ・ `finalCallback` は、このページ上のコールバック関数の名前です。
- ・ `pendingCallback` (オプション) は、このページ上のもう 1 つのコールバック関数の名前です。
- ・ `filters` は、クエリに適用する追加のフィルタを指定します。

このメソッドは、指定された MDX クエリを実行します。クエリが保留中のときや完了したときに、システムは指定されたコールバック関数を呼び出します。

このメソッドには戻り値はありません。

`runMDXDrillQuery()`

```
runMDXDrillQuery(mdx,finalCallback,pendingCallback,listing,fieldList,filters)
```

以下はその説明です。

- ・ `mdx` は MDX SELECT クエリです。
- ・ `finalCallback` は、このページ上のコールバック関数の名前です。
- ・ `pendingCallback` (オプション) は、このページ上のもう 1 つのコールバック関数の名前です。
- ・ `listing` は詳細リストの名前です。
- ・ `fieldList` は、リスト・フィールドのリストを指定します。`listing` または `fieldList` を指定します。
- ・ `filters` は、クエリに適用する追加のフィルタを指定します。

このメソッドは、指定されたドリルスルー・クエリを実行します。クエリが保留中のときや完了したときに、システムは指定されたコールバック関数を呼び出します。

このメソッドには戻り値はありません。

`runPivot()`

```
runPivot(pivot,finalCallback,pendingCallback,filters)
```

以下はその説明です。

- ・ `pivot` は、ピボット・テーブルの論理名です。
- ・ `finalCallback` は、このページ上のコールバック関数の名前です。
- ・ `pendingCallback` (オプション) は、このページ上のもう 1 つのコールバック関数の名前です。
- ・ `filters` は、クエリに適用する追加のフィルタを指定します。

このメソッドは、指定されたピボット・テーブルで定義されている MDX クエリを実行します。クエリが保留中のときや完了したときに、システムは指定されたコールバック関数を呼び出します。

このメソッドには戻り値はありません。

DeepSeeUtils

Business Intelligence を操作するための追加のメソッドを提供します。

このオブジェクトを使用できる場所

このオブジェクトは、クライアント側の JavaScript コード内で使用できます（そのコードで **DeepSee.js** がインクルードされている場合）。“[DeepSee JavaScript API の概要](#)” を参照してください。

このオブジェクトの作成

クライアント・コード内で **DeepSee.js** をインクルードすると、DeepSeeUtils オブジェクトが自動的に使用可能になります。

このオブジェクトのメソッド

DeepSeeUtils オブジェクトには、以下のメソッドが用意されています。

getCubeList()

```
getCubeList(connection, finalFunc)
```

以下はその説明です。

- ・ connection — Business Intelligence [データ・コネクタ・オブジェクト](#)。
- ・ finalFunc — 結果が得られたときに実行する関数の名前を指定します。

このメソッドは、使用可能なキューブに関する情報を取得します。これは、[POST /Info/Cubes](#) REST 呼び出しを行います。この呼び出しの応答オブジェクトは、finalFunc で指定された関数で使用できます。

getCubeListings()

```
getCubeListings(connection, finalFunc)
```

以下はその説明です。

- ・ connection — Business Intelligence [データ・コネクタ・オブジェクト](#)。
- ・ cubename — キューブの論理名を指定します。
- ・ finalFunc — 結果が得られたときに実行する関数の名前を指定します。

このメソッドは、指定されたキューブで使用するリストに関する情報を取得します。これは、[POST /Info/Listings/:cube](#) REST 呼び出しを行います。この呼び出しの応答オブジェクトは、finalFunc で指定された関数で使用できます。

getDashboardList()

```
getDashboardList(connection, finalFunc)
```

このメソッドは、使用可能なダッシュボードに関する情報を取得します。これは、[POST /Info/Dashboards](#) REST 呼び出しを行います。この呼び出しの応答オブジェクトは、finalFunc で指定された関数で使用できます。

getErrorMessage()

```
getErrorMessage(data)
```

data は、いずれかの Business Intelligence [REST サービス](#)によって返される応答オブジェクトです。

このメソッドは、そのオブジェクトに含まれているエラー・メッセージ (ある場合) のテキストを返します。

getFiltersForDataSource()

```
getFiltersForDataSource(connection, cubename, finalcallback)
```

以下はその説明です。

- ・ connection – Business Intelligence [データ・コネクタ・オブジェクト](#)。
- ・ cubename – キューブの論理名を指定します。
- ・ finalcallback – 結果が得られたときに実行する関数の名前を指定します。

このメソッドは、指定されたキューブで利用できるフィルタに関する情報を取得します。これは、[POST /Info/Filters/:datasource](#) REST 呼び出しを行います。この呼び出しの応答オブジェクトは、finalFunc で指定された関数で使用できます。

getMembersForFilter()

```
getMembersForFilter(connection, cubeName, filterSpec, finalCallback)
```

以下はその説明です。

- ・ connection – Business Intelligence [データ・コネクタ・オブジェクト](#)。
- ・ cubename – キューブの論理名を指定します。
- ・ filterSpec – 取得するメンバが属するフィルタを指定します。
- ・ finalcallback – 結果が得られたときに実行する関数の名前を指定します。

このメソッドは、指定されたフィルタのメンバに関する情報を取得します。これは、[POST /Info/FilterMembers/:datasource/:filterSpec](#) REST 呼び出しを行います。この呼び出しの応答オブジェクトは、finalFunc で指定された関数で使用できます。

getPivotList()

```
getPivotList(connection, finalFunc)
```

以下はその説明です。

- ・ connection – Business Intelligence [データ・コネクタ・オブジェクト](#)。
- ・ finalFunc – 結果が得られたときに実行する関数の名前を指定します。

このメソッドは、使用可能なピボット・テーブルに関する情報を取得します。これは、[POST /Info/Pivots](#) REST 呼び出しを行います。この呼び出しの応答オブジェクトは、finalFunc で指定された関数で使用できます。

getResultsAsArray()

```
getResultsAsArray(data)
```

data は、いずれかの Business Intelligence [REST サービス](#)によって返される応答オブジェクトです。

このメソッドは、以下の規則に従ってそのオブジェクト内の結果の配列を返します。

1. 応答オブジェクトに `Result.Filters` 配列が含まれている場合は、このメソッドはその配列を返します。
2. 応答オブジェクトに `Result.FilterMembers` 配列が含まれている場合は、このメソッドはその配列を返します。

3. 応答オブジェクトに `Result.Listings` 配列が含まれている場合は、このメソッドはその配列を返します。

上記のいずれにも該当しない場合は、このメソッドは何も返しません。

`isError()`

`isError(data)`

`data` は、いずれかの Business Intelligence [REST サービス](#)によって返される応答オブジェクトです。

エラーが発生したことが応答オブジェクトによって示されている場合は、このメソッドは `true` を返します。そうでない場合は、`false` を返します。

Business Intelligence REST API

このリファレンス・セクションでは、InterSystems IRIS® データ・プラットフォーム [Business Intelligence](#) の REST サービスに関する情報を提供します。これらのサービスは、%DeepSee.REST.v1 クラスによって提供されます。["Business Intelligence REST API の使用法"](#) を参照してください。

GET /Data/GetDSTIME

特定のキューブに対してサーバが処理した最後の ^OBJ.DSTIME タイムスタンプを取得します。

URL パラメータ

sourceClass	必須項目。キューブのソース・クラスのフルネーム。
-------------	--------------------------

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- ・ 要求のメソッド :

GET

- ・ 要求の URL :

`https://localhost/api/deepsee/v1/Data/GetDSTime/HoleFoods.Transation`

URL の有効な形式に関する説明は、"[Business Intelligence REST API の概要](#)" を参照してください。

応答の例

```
{
  "Status": "OK",
  "DispatchClass": "%DeepSee.REST.v1.InfoServer",
  "NameSpace": "SAMPLES"
}
```


GET /Data/TestConnection

サーバへの接続をテストします。

URL パラメータ

なし。

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- ・ 要求のメソッド :

GET

- ・ 要求の URL :

`https://localhost/api/deepsee/v1/Data/TestConnection`

URL の有効な形式に関する説明は、["Business Intelligence REST API の概要"](#) を参照してください。

応答の例

```
{
  "Status": "OK",
  "DispatchClass": "%DeepSee.REST.v1.DataServer",
  "NameSpace": "SAMPLES"
}
```

GET /Info/TestConnection

サーバへの接続をテストします。

URL パラメータ

なし。

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- ・ 要求のメソッド :

GET

- ・ 要求の URL :

`https://localhost/api/deepsee/v1/Info/TestConnection`

URL の有効な形式に関する説明は、["Business Intelligence REST API の概要"](#) を参照してください。

応答の例

```
{
  "Status": "OK",
  "DispatchClass": "%DeepSee.REST.v1.InfoServer",
  "NameSpace": "SAMPLES"
}
```

POST /Command/BuildAllRegisteredGroups

現在のキューブ・レジストリに登録されているすべてのキューブを構築します。

URL パラメータ

なし。

要求の本文の詳細

このサービスは空の本文で呼び出されます。

要求の例

- 要求のメソッド :

POST

- 要求の URL :

`https://localhost/api/deepsee/v2/Command/BuildAllRegisteredGroups`

URL の有効な形式に関する説明は、"[Business Intelligence REST API の概要](#)" を参照してください。

- 要求の本文 :

```
{}
```

応答の例

```
{
  "buildStats": {
    "buildStats_1": {
      "caption": "Filter RDoctors",
      "elapsedTime": 0.01413,
      "errors": 0,
      "event": "build",
      "eventId": "124",
      "expressionTime": 0.000039,
      "factCount": 40,
      "iKnowTime": 0,
      "missingReferences": 0,
      "status": 1
    },
    "buildStats_2": {
      "caption": "Filter RPatients",
      "elapsedTime": 0.148839,
      "errors": 0,
      "event": "build",
      "eventId": "126",
      "expressionTime": 0.053942,
      "factCount": 1000,
      "iKnowTime": 0,
      "missingReferences": 0,
      "status": 1
    },
    "buildStats_3": {
      "caption": "Patients",
      "elapsedTime": 0.490752,
      "errors": 0,
      "event": "build",
      "eventId": "122",
      "expressionTime": 0.343703,
      "factCount": 1000,
      "iKnowTime": 0,
      "missingReferences": 0,
      "status": 1
    },
    "buildStats_4": {
      "caption": "RCities",
      "elapsedTime": 0.010991,
```

```

        "errors": 0,
        "event": "build",
        "eventId": "123",
        "expressionTime": 0.000003,
        "factCount": 9,
        "iKnowTime": 0,
        "missingReferences": 0,
        "status": 1
    },
    "buildStats_5": {
        "caption": "RCityRainfall",
        "elapsedTime": 0.006289,
        "errors": 0,
        "event": "build",
        "eventId": "127",
        "expressionTime": 0,
        "factCount": 0,
        "iKnowTime": 0,
        "missingReferences": 0,
        "status": 1
    },
    "buildStats_6": {
        "caption": "RDoctors",
        "elapsedTime": 0.014014,
        "errors": 0,
        "event": "build",
        "eventId": "125",
        "expressionTime": 0.000041,
        "factCount": 40,
        "iKnowTime": 0,
        "missingReferences": 0,
        "status": 1
    },
    "buildStats_7": {
        "caption": "RPatients",
        "elapsedTime": 0.136887,
        "errors": 0,
        "event": "build",
        "eventId": "128",
        "expressionTime": 0.051799,
        "factCount": 1000,
        "iKnowTime": 0,
        "missingReferences": 0,
        "status": 1
    },
    "buildStats_8": {
        "caption": "RPatientsDependsOn",
        "elapsedTime": 0.146196,
        "errors": 0,
        "event": "build",
        "eventId": "129",
        "expressionTime": 0.051907,
        "factCount": 1000,
        "iKnowTime": 0,
        "missingReferences": 0,
        "status": 1
    },
    "buildStats_9": {
        "caption": "RPatientsInherit",
        "elapsedTime": 0.163897,
        "errors": 0,
        "event": "build",
        "eventId": "130",
        "expressionTime": 0.059377,
        "factCount": 1000,
        "iKnowTime": 0,
        "missingReferences": 0,
        "status": 1
    }
},
"Error": "",
"status": 1
}

```

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#)を参照してください。

POST /Command/BuildCube

ターゲット・キューブを構築し、イベント情報を記録および報告します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

cubeName	必須項目。構築するキューブの名前。
----------	-------------------

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v2/Command/BuildCube`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文：

```
{
    "cubeName": "Patients"
}
```

応答の例

```
{
    "buildStats": {
        "caption": "Patients",
        "elapsedTime": 0.594205,
        "errors": 0,
        "event": "build",
        "eventId": "119",
        "expressionTime": 0.396416,
        "factCount": 1000,
        "iKnowTime": 0,
        "missingReferences": 0,
        "status": 1
    },
    "Error": "",
    "status": 1
}
```

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Command/BuildOneRegisteredGroup

登録されているキューブ・グループ内のすべてのキューブをリストされている順序で構築し、構築プロセスに関する統計を返します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

groupName	必須項目。構築するキューブ・グループの名前。
-----------	------------------------

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v2/Command/BuildOneRegisteredGroup`

URL の有効な形式に関する説明は、["Business Intelligence REST API の概要"](#) を参照してください。

- 要求の本文：

```
{
    "groupName": "Group 1"
}
```

応答の例

```
{
    "buildStats": {
        "caption": "Patients",
        "elapsedTime": 0.594205,
        "errors": 0,
        "event": "build",
        "eventId": "119",
        "expressionTime": 0.396416,
        "factCount": 1000,
        "iKnowTime": 0,
        "missingReferences": 0,
        "status": 1
    },
    "Error": "",
    "status": 1
}
```

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#) を参照してください。

POST /Command/BuildRegistryMap

キューブ・レジストリと、システム上のすべての未登録キューブの現在の状態のマップを構築します。

URL パラメータ

なし。

要求の本文の詳細

このサービスは空の本文で呼び出されます。

要求の例

- 要求のメソッド :
POST
- 要求の URL :
`https://localhost/api/deepsee/v2/Command/BuildRegistryMap`
URL の有効な形式に関する説明は、"[Business Intelligence REST API の概要](#)" を参照してください。
- 要求の本文 :
`{}`

応答の例

```
{
  "registryMap": {
    "is a sample Cube Registry for QD testing\"},
    "lookup": {
      "cubes": [
        {
          "FILTERRDOCTORS": [
            2,
            2
          ],
          "FILTERRPATIENTS": [
            2,
            4
          ],
          "PATIENTS": [
            1,
            1
          ],
          "RCITIES": [
            2,
            1
          ],
          "RCITYRAINFALL": [
            2,
            5
          ],
          "RDOCTORS": [
            2,
            3
          ],
          "RPATIENTS": [
            2,
```

```
        6
      ],
      "RPATIENTSDEPENDSON": [
        2,
        7
      ],
      "RPATIENTSINHERIT": [
        2,
        8
      ]
    }
  ],
  "groups": [
    {
      "Group 1": [
        1
      ],
      "Group 2": [
        2
      ]
    }
  ]
},
"Error": "",
"status": 1
}
```

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#)を参照してください。

POST /Command/GetActiveRegistry

登録されているキューブ・グループ内のすべてのキューブをリストされている順序で構築し、構築プロセスに関する統計を返します。

URL パラメータ

なし。

要求の本文の詳細

このサービスは空の本文で呼び出されます。

要求の例

- ・ 要求のメソッド :

POST

- ・ 要求の URL :

`https://localhost/api/deepsee/v2/Command/GetActiveRegistry`

URL の有効な形式に関する説明は、"[Business Intelligence REST API の概要](#)" を参照してください。

- ・ 要求の本文 :

```
{}
```

応答の例

```
{
  "activeRegistry": "DeepSee.CubeManager.CubeRegistryDefinition",
  "Error": "",
  "status": 1
}
```

すべての応答オブジェクトに適用される情報は、"[応答オブジェクトに関する注意事項](#)" を参照してください。

POST /Command/GetCubeSize

キューブの名前を指定すると、そのファクト・テーブル内の行の数を返します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

cubeKey	必須項目。対象となるサイズを持つキューブの論理名。
---------	---------------------------

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v2/Command/GetCubeSize`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文：

```
{
  "cubeKey": "Patients"
}
```

応答の例

```
{
  "cubeSize": 1000,
  "Error": "",
  "status": 1
}
```

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Command/GetLastUpdate

キューブの登録データの最終更新日時のタイムスタンプを取得します。キューブ・マネージャの外部で実行された更新は、このチェックでは認識されません。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

cubeKey	必須項目。最終更新日時を取得するキューブの論理名。
---------	---------------------------

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v2/Command/GetLastUpdate`

URL の有効な形式に関する説明は、["Business Intelligence REST API の概要"](#) を参照してください。

- 要求の本文：

```
{
    "groupName": "Group 1"
}
```

応答の例

```
{
    "tSC": 1,
    "lastUpdate": "2020-04-30 16:58:00",
    "Error": "",
    "status": 1
}
```

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#) を参照してください。

POST /Command/getSynchScheduleParameters

%SYS.Task.DeepSeeSynchronize のパラメータを取得および表示する便利なメソッドです。

URL パラメータ

なし。

要求の本文の詳細

このサービスは空の本文で呼び出されます。

要求の例

- ・ 要求のメソッド :

POST

- ・ 要求の URL :

`https://localhost/api/deepsee/v2/Command/getSynchScheduleParameters`

URL の有効な形式に関する説明は、"[Business Intelligence REST API の概要](#)" を参照してください。

- ・ 要求の本文 :

`{}`

応答の例

```
{
  "synchParams": {
    "dailyFrequency": 1,
    "dailyFrequencyTime": 0,
    "dailyIncrement": 32,
    "dailyStartTime": 0
  },
  "Error": "",
  "status": 1
}
```

すべての応答オブジェクトに適用される情報は、"[応答オブジェクトに関する注意事項](#)" を参照してください。

POST /Command/IsValidCubeSchedule

現在のキューブ設定に対して検証を実行します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

必須項目。検証するスケジュールを持つキューブの名前。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v2/Command/IsValidCubeSchedule`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文：

```
{
  "cube": "HoleFoods Base"
}
```

応答の例

```
{
  "isValid": 1,
  "Error": "",
  "status": 1
}
```

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Command/IsValidGroup

指定されたグループがメンバの自然なグループの集合と同等かどうかをテストします。true の場合には 1 を返します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

group	必須項目。テストするキューブ・グループの名前。
-------	-------------------------

要求の例

- ・ 要求のメソッド：

POST

- ・ 要求の URL :

https://localhost/api/deepsee/v2/Command/IsValidGroup

URL の有効な形式に関する説明は、"[Business Intelligence REST API の概要](#)"を参照してください。

- ・ 要求の本文：

```
{
    "groupName": "HoleFoods Base"
}
```

応答の例

[illegible]

```

        ]
      }
    ],
    "5": [
      {
        "MYCUBE89212": [
          1
        ]
      }
    ],
    "6": [
      {
        "NEWHOLEFOODS": [
          1
        ]
      }
    ],
    "7": [
      {
        "PATIENTS": [
          1
        ]
      }
    ],
    "8": [
      {
        "PREBUILTHOLEFOODS": [
          1
        ]
      }
    ],
    "9": [
      {
        "FILTERRDOCTORS": [
          2
        ],
        "FILTERRPATIENTS": [
          5
        ],
        "RCITIES": [
          1
        ],
        "RCITYRAINFALL": [
          3
        ],
        "RDOCTORS": [
          4
        ],
        "RPATIENTS": [
          6
        ],
        "RPATIENTSDEPENDSON": [
          7
        ],
        "RPATIENTSINHERIT": [
          8
        ]
      }
    ]
  ],
  "cubes": {
    "DEPENDSONHOLEFOODS": [
      1
    ],
    "EXPRESSIONHOLEFOODS": [
      2
    ],
    "FILTERRDOCTORS": [
      9,
      {
        "dependents": [
          2,
          {
            "FILTERRPATIENTS": [],
            "RPATIENTS": []
          }
        ]
      }
    ],
    "FILTERRPATIENTS": [
      9
    ],
    "HOLEFOODS": [
      3
    ]
  }
}

```

```

    ],
    "JMD1340": [
      4
    ],
    "MYCUBE89212": [
      5
    ],
    "NEWHOLEFOODS": [
      6
    ],
    "PATIENTS": [
      7
    ],
    "PREBUILTHOLEFOODS": [
      8
    ],
    "RCITIES": [
      9,
      {
        "dependents": [
          7,
          {
            "FILTERRDOCTORS": [],
            "FILTERRPATIENTS": [],
            "RCITYRAINFALL": [],
            "RDOCTORS": [],
            "RPATIENTS": [],
            "RPATIENTSDEPENDSON": [],
            "RPATIENTSINHERIT": []
          }
        ]
      }
    ],
    "RCITYRAINFALL": [
      9
    ],
    "RDOCTORS": [
      9,
      {
        "dependents": [
          3,
          {
            "RPATIENTS": [],
            "RPATIENTSDEPENDSON": [],
            "RPATIENTSINHERIT": []
          }
        ]
      }
    ],
    "RPATIENTS": [
      9
    ],
    "RPATIENTSDEPENDSON": [
      9
    ],
    "RPATIENTSINHERIT": [
      9
    ]
  },
  "buildOrders": {
    "1": [
      1,
      {
        "1": [
          "DEPENDSONHOLEFOODS"
        ]
      }
    ],
    "2": [
      1,
      {
        "1": [
          "EXPRESSIONHOLEFOODS"
        ]
      }
    ],
    "3": [
      1,
      {
        "1": [
          "HOLEFOODS"
        ]
      }
    ],
    "4": [

```



```

        1,
        {
            "1": [
                "JMD1340"
            ]
        }
    ],
    "5": [
        1,
        {
            "1": [
                "MYCUBE89212"
            ]
        }
    ],
    "6": [
        1,
        {
            "1": [
                "NEWHOLEFOODS"
            ]
        }
    ],
    "7": [
        1,
        {
            "1": [
                "PATIENTS"
            ]
        }
    ],
    "8": [
        1,
        {
            "1": [
                "PREBUILTHOLEFOODS"
            ]
        }
    ],
    "9": [
        8,
        {
            "1": [
                "RCITIES"
            ],
            "2": [
                "FILTERRDOCTORS"
            ],
            "3": [
                "RCITYRAINFALL"
            ],
            "4": [
                "RDOCTORS"
            ],
            "5": [
                "FILTERRPATIENTS"
            ],
            "6": [
                "RPATIENTS"
            ],
            "7": [
                "RPATIENTSDEPENDSON"
            ],
            "8": [
                "RPATIENTSINHERIT"
            ]
        }
    ]
    ],
    "Error": "",
    "status": 1
}

```

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#)を参照してください。

POST /Command/RepairBuild

登録済みキューブの計画外の修復構築を実行します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

cubeName	必須項目。修復するキューブの名前。
----------	-------------------

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v2/Command/RepairBuild`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文：

```
{
  "cubeName": "Patients"
}
```

応答の例

```
{
  "buildStats": {
    "buildStats_1": {
      "caption": "Patients",
      "elapsedTime": 0.494022,
      "errors": 0,
      "event": "build",
      "eventId": "120",
      "expressionTime": 0.345886,
      "factCount": 1000,
      "iKnowTime": 0,
      "missingReferences": 0,
      "status": 1
    }
  },
  "Error": "",
  "status": 1
}
```

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Command/SetActiveRegistry

現在アクティブなキューブ・レジストリを変更します。

URL パラメータ

なし。

要求の本文の詳細

このサービスは空の要求の本文を使用します。

要求の例

- ・ 要求のメソッド :

POST

- ・ 要求の URL :

`https://localhost/api/deepsee/v2/Command/setActiveRegistry`

URL の有効な形式に関する説明は、"[Business Intelligence REST API の概要](#)" を参照してください。

- ・ 要求の本文 :

```
{}
```

応答の例

```
{
```

この API 呼び出しは空の応答を返しますが、`^DeepSee.CubeManager("activeRegistry")` を `DeepSee.CubeManager.CubeRegistryDefinition` に設定します。

すべての応答オブジェクトに適用される情報は、"[応答オブジェクトに関する注意事項](#)" を参照してください。

POST /Command/SynchronizeCube

%DeepSee.Utils.%SynchronizeCube のラップです。キューブの構築時にキューブのイベント情報を記録します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

cubeName	必須項目。同期するキューブの名前。
----------	-------------------

要求の例

- 要求のメソッド：

POST

- 要求の URL：

https://localhost/api/deepsee/v2/Command/SynchronizeCube

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文：

```
{
    "cubeName": "Patients"
}
```

応答の例

```
{
    "factsUpdated": 0,
    "synchronizeStats": {
        "caption": "Patients",
        "event": "synchronize",
        "eventId": "132",
        "status": 1
    },
    "Error": "",
    "status": 1
}
```

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Command/WriteToRegistry

CubeManager.RegistryMap オブジェクトを検索し、すべてのキューブを登録します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

cubeMap	必須項目。検索する RegistryMap オブジェクトの名前。
---------	----------------------------------

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v2/Command/WriteToRegistry`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文：

```
{
    "cubeMap": "Groups"
}
```

応答の例

```
{
    "cubeMap":
    "{ \"Groups\": [ { \"Cubes\": [ { \"CubeKey\": \"EXPRESSIONHOLEFOODS\", \"CubeDisplayName\": \"HoleFoods
    \"Error\": \"\",
    \"status\": 1,
    \"tSC\": 1
  }
}
```

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Data/KPIExecute

KPI で定義されたクエリを実行します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

KPI	必須項目。KPI の論理名。
FILTERS	<p>オプション。KPI のフィルタ処理方法を指定するフィルタ・オブジェクトの配列。各フィルタ・オブジェクトは、以下のプロパティを提供する必要があります。</p> <ul style="list-style-type: none"> name – フィルタの仕様 ([aged].[h1].[age group] など) value – フィルタのメンバの論理名 ([0 to 29] など)

要求の例

- 要求のメソッド：

POST

- 要求の URL：

https://localhost/api/deepsee/v1/Data/KPIExecute

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)” を参照してください。

- 要求の本文：

```
{
  "KPI": "DemoMDX",
  "FILTERS": [ { "name" : "[aged].[h1].[age group]", "value" : "&[0 to 29]" } ]
}
```

応答の例

```
{
  "Info": { "Error": "", "KpiName": "DemoMDX" },
  "Result": {
    "Properties": [
      { "name": "PatCount", "caption": "PatCount", "columnNo": 1 },
      { "name": "AvgTestScore", "caption": "AvgTestScore", "columnNo": 2 }
    ],
    "Series": [
      { "PatCount": 482, "AvgTestScore": 73.62564102564102564, "seriesName": "Cedar Falls" },
      { "PatCount": 473, "AvgTestScore": 74.54089709762532982, "seriesName": "Centerville" },
      { "PatCount": 454, "AvgTestScore": 73.86532951289398281, "seriesName": "Cypress" },
      { "PatCount": 471, "AvgTestScore": 75.69459459459459459, "seriesName": "Elm Heights" },
      { "PatCount": 468, "AvgTestScore": 74.00806451612903226, "seriesName": "Juniper" },
      { "PatCount": 464, "AvgTestScore": 73.71925133689839572, "seriesName": "Magnolia" },
      { "PatCount": 438, "AvgTestScore": 73.76123595505617978, "seriesName": "Pine" },
      { "PatCount": 464, "AvgTestScore": 75.46537396121883657, "seriesName": "Redwood" },
      { "PatCount": 445, "AvgTestScore": 75.19886363636363636, "seriesName": "Spruce" }
    ]
  }
}
```

```
    }  
  }  
}
```

この応答オブジェクト内では、Result プロパティに Series プロパティと Properties プロパティが含まれています。Series プロパティにはオブジェクトの配列が含まれており、KPI 内の系列（または行）ごとに 1 つのオブジェクトがあります。Properties プロパティにはオブジェクトの配列が含まれており、KPI 内の行ごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#)を参照してください。

POST /Data/MDXCancelQuery/:queryID

指定されたクエリ ID を持つ開始済みのクエリを取り消します。

URL パラメータ

queryID	必須項目。取り消すクエリの ID。 POST /Data/MDXExecute サービスを使用してこのクエリを開始した場合は、このサービスによって返された応答オブジェクトの <code>Info.QueryID</code> プロパティからこのクエリの ID を取得します。 POST /Data/PivotExecute サービスを使用してこのクエリを開始した場合は、このサービスによって返された応答オブジェクトの <code>Info.QueryID</code> プロパティからこのクエリの ID を取得します。
---------	---

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Data/MDXCancelQuery/:patients||en2515296118`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

POST /Data/MDXDrillthrough

詳細リストを実行します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

MDX	<p>必須項目。MDX SELECT クエリ。その前方に DRILLTHROUGH または DRILLFACTS を指定します。</p> <p>指定した詳細リストを使用するには、またはキューブのソース・クラスからフィールドを取得するには、DRILLTHROUGH を指定します。ファクト・テーブルからフィールドを取得するには、DRILLFACTS を指定します。</p> <p>ベース SELECT クエリから複数のセルが返される場合は、左上のセルのみが詳細リスト用に使用されます。</p>
LISTING	<p>オプション。使用する詳細リストの論理名。LISTING または RETURN を指定する必要がありますが、両方を指定してはいけません。</p>
WAIT	<p>オプション。0 または 1 (既定値) を指定します。このプロパティの値が 0 の場合は、サーバは部分的な結果を送信します。このプロパティの値が 1 の場合、サーバ側では、クライアントが完全な結果が得られるまで待ってから応答を送信することを希望していると想定します。</p>
RETURN	<p>オプション。MDX の値に応じて、該当するテーブル内のフィールドのリスト。このプロパティを指定した場合は、リストはこれらのフィールドで構成されます。</p>

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Data/MDXDrillthrough`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文：

```
{
  "MDX" : "DRILLTHROUGH SELECT FROM patients WHERE AGED.60",
  "LISTING" : "Patient details"
}
```

別の例を示します。

```
{
  "MDX" : "DRILLTHROUGH SELECT FROM patients WHERE AGED.60",
  "RETURN" : "Age,BirthDate"
}
```

別の例を示します。

```
{
  "MDX" : "DRILLFACTS SELECT FROM patients WHERE AGED.60",
  "RETURN": "MxAge,MxTestScore"
}
```

応答の例

```
{
  "Info": {
    "Error": "",
    "MDXText": "DRILLTHROUGH SELECT FROM [PATIENTS] WHERE [AGED].[60]",
    "QueryKey": "en2156087935", "CubeKey": "PATIENTS",
    "QueryID": "PATIENTS|en2156087935", "Cube": "patients",
    "ResultsComplete": 1, "Pivot": "", "QueryType": "DRILLTHROUGH", "ListingSource": "source",
    "ColCount": 5, "RowCount": 0, "Error": "", "TimeStamp": "2016-08-14 15:43:04"
  },
  "AxesInfo": [
    { "%ID": "SlicerInfo", "Text": "[AGED].[60]" },
    { "%ID": "AxisInfo_1", "Text": "[%SEARCH]" },
    { "%ID": "AxisInfo_2", "Text": "[%SEARCH]" }
  ],
  "Result": { "children":
    [
      { "PatientID": "SUBJ_100508", "Age": 60, "Gender": "Female", "Home City": "Elm Heights", "Test Score": 81 },
      { "PatientID": "SUBJ_100539", "Age": 60, "Gender": "Female", "Home City": "Elm Heights", "Test Score": 90 },
      { "PatientID": "SUBJ_100701", "Age": 60, "Gender": "Female", "Home City": "Redwood", "Test Score": 61 },
      { "PatientID": "SUBJ_100829", "Age": 60, "Gender": "Female", "Home City": "Juniper", "Test Score": 98 },
      ... ]
    }
  }
}
```

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#)を参照してください。

POST /Data/MDXExecute

MDX クエリを実行して、結果を取得します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

MDX	必須項目。MDX SELECT クエリ。
FILTERS	オプション。このクエリに追加するフィルタ。指定されている場合、このプロパティの値は文字列の配列にする必要があり、それぞれの文字列ではフィルタ値を指定します。
WAIT	オプション。0 または 1 (既定値) を指定します。このプロパティの値が 0 の場合は、サーバは部分的な結果を送信します。このプロパティの値が 1 の場合、サーバ側では、クライアントが完全な結果が得られるまで待ってから応答を送信することを希望していると想定します。
TIMEOUT	オプション。クエリ結果を待つためのタイムアウト時間 (秒単位)。この待機タイムアウト時間の既定値は、セッションのタイムアウト設定値から 2 秒を差し引いた秒数です。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Data/MDXExecute`

URL の有効な形式に関する説明は、["Business Intelligence REST API の概要"](#) を参照してください。

- 要求の本文：

```
{"MDX": "SELECT aged.[age group].MEMBERS ON 0 FROM PATIENTS"}
```

別の例を示します。

```
{"MDX": "SELECT FROM PATIENTS"}
```

別の例を示します。

```
{"MDX": "SELECT birthd.date.members on 0 FROM PATIENTS", "WAIT":1, "TIMEOUT":30}
```

別の例を示します。

```
{
  "MDX": "SELECT FROM PATIENTS",
  "FILTERS": [ "[HomeD].[H1].[ZIP].&[32006]", "[HomeD].[H1].[ZIP].&[32007]" ],
  "WAIT":1,
  "TIMEOUT":30
}
```

応答の例

MDX クエリの応答

POST /Data/PivotExecute と POST /Data/MDXUpdateResults は同じ応答本文を返します。

Info.QueryID プロパティに含まれているクエリ ID は、POST /Data/MDXCancelQuery サービスと POST /Data/MDXUpdateResults サービスの入力として必要です。Info.ResultsComplete プロパティの値が 1 の場合は、MDX クエリが完了していることを示します。Info.PendingResults プロパティの値が 1 の場合、プラグインはまだ計算処理中です。ただし、クエリの他の部分は完了している可能性があります。Info.PendingResults プロパティの値が 0 の場合は、プラグインが計算を終了していることを示します。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

```
{
  "Info": {
    "Error": "",
    "MDXText": "SELECT [AGED].[AGE GROUP].MEMBERS ON 0 FROM [PATIENTS]",
    "QueryKey": "en2772997983",
    "CubeKey": "PATIENTS",
    "QueryID": "PATIENTS|en2772997983",
    "Cube": "PATIENTS",
    "ResultsComplete": 1,
    "Pivot": "",
    "QueryType": "SELECT",
    "ListingSource": "",
    "ColCount": 3,
    "RowCount": 0,
    "TimeStamp": "2016-08-14 16:05:16"
  },
  "AxesInfo": [
    { "%ID": "SlicerInfo", "Text": "" },
    { "%ID": "AxisInfo_1", "Text": "[AGED].[AGE GROUP].MEMBERS" },
    { "%ID": "AxisInfo_2", "Text": "[%SEARCH]" }
  ],
  "Result": {
    "Axes": [
      { "%ID": "Axis_1", "Tuples": [
        { "%ID": "Tuple_1",
          "Members": [
            { "%ID": "Member_1", "Name": "0 to 29" }
          ],
          "MemberInfo": [
            { "%ID": "MemberInfo_1",
              "nodeNo": 3, "text": "0 to 29",
              "dimName": "AgeD",
              "hierName": "H1",
              "levelName": "Age Group",
              "memberKey": "0 to 29",
              "dimNo": 2,
              "hierNo": 1,
              "levelNo": 2,
              "aggregate": "",
              "orSpec": "" }
            ]
          },
        { "%ID": "Tuple_2", ... },
        { "%ID": "Tuple_3", ... }
      ]
    },
    "TupleInfo": [
      { "%ID": "TupleInfo_1", "childSpec": "[AgeD].[H1].[Age Group].&[0 to 29].children" },
      { "%ID": "TupleInfo_2", "childSpec": "[AgeD].[H1].[Age Group].&[30 to 59].children" },
      ...
    ],
    "CellData": [
      { "%ID": "Cell_1", "ValueLogical": 418, "Format": "", "ValueFormatted": "418" },
      { "%ID": "Cell_2", "ValueLogical": 421, "Format": "", "ValueFormatted": "421" },
      ...
    ]
  }
}
```

応答が不完全な場合は、応答には次のようなセル・データ・オブジェクトが含まれます。

```
{ "%ID": "Cell_9", "ValueLogical": "@Computing
0.00%", "Format": "", "ValueFormatted": "@Computing 0.00%" }
```

MDX DRILLTHROUGH クエリの応答

```

{
  "Info": {
    "Error": "",
    "TimeStamp": "2017-09-26 15:31:23",
    "ResultsComplete": 1,
    "MDXText": "DRILLTHROUGH SELECT [AGED].[AGE GROUP].[0 TO 29] ON 0 FROM [PATIENTS]",

    "QueryKey": "en2983351588",
    "CubeKey": "PATIENTS",
    "QueryID": "PATIENTS|en2983351588",
    "Cube": "PATIENTS",
    "Pivot": "",
    "QueryType": "DRILLTHROUGH",
    "ListingSource": "source",
    "ColCount": 5,
    "RowCount": 0
  },
  "AxesInfo": [
    { "%ID": "SlicerInfo",
      "Text": "" },
    { "%ID": "AxisInfo_1",
      "Text": "[AGED].[AGE GROUP].[0 TO 29]" },
    { "%ID": "AxisInfo_2",
      "Text": "[%SEARCH]" }
  ],
  "Result": {
    "children": [
      { "PatientID": "SUBJ_100786",
        "Age": 0,
        "Gender": "Female",
        "Home City": "Centerville",
        "Test Score": 77 },
      { "PatientID": "SUBJ_100960",
        "Age": 0,
        "Gender": "Female",
        "Home City": "Elm Heights",
        "Test Score": 62 },
      { "PatientID": "SUBJ_100977",
        "Age": 0,
        "Gender": "Female",
        "Home City": "Elm Heights",
        "Test Score": 54 },
      ... ]
    }
  }
}

```

POST /Data/MDXUpdateResults/:queryID

以前は不完全であった指定されたクエリの更新された結果を取得します。

URL パラメータ

queryID	必須項目。クエリの ID。 POST /Data/MDXExecute サービスを使用してこのクエリを開始した場合は、このサービスによって返された応答オブジェクトの <code>Info.QueryID</code> プロパティからこのクエリの ID を取得します。 POST /Data/PivotExecute サービスを使用してこのクエリを開始した場合は、このサービスによって返された応答オブジェクトの <code>Info.QueryID</code> プロパティからこのクエリの ID を取得します。
---------	---

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Data/MDXCancelQuery/:patients||en2515296118`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)” を参照してください。

応答の例

このサービスは、[POST /Data/MDXExecute](#) と同じ応答オブジェクトを返します。

POST /Data/PivotExecute

ピボット・テーブルで定義された MDX クエリを実行して、結果を取得します。

URL パラメータ

なし。要求の本文は必須です。次のセクションを参照してください。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

PIVOT	必須項目。ピボット・テーブルの完全な論理名 (ピボット・テーブルが格納されているフォルダの名前を含む)。
FILTERS	オプション。このクエリに追加するフィルタを指定します。指定されている場合、このプロパティの値は文字列の配列にする必要があり、それぞれの文字列ではフィルタ値を指定します。
WAIT	オプション。0 または 1 (既定値) を指定します。このプロパティの値が 0 の場合は、サーバは部分的な結果を送信します。このプロパティの値が 1 の場合、サーバ側では、クライアントが完全な結果が得られるまで待ってから応答を送信することを希望していると想定します。
TIMEOUT	オプション。クエリ結果を待つためのタイムアウト時間 (秒単位)。この待機タイムアウト時間の既定値は、セッションのタイムアウト設定値から 2 秒を差し引いた秒数です。
VARIABLES	オプション。ピボット・テーブルで使用されているピボット変数の値を指定します。これは、オブジェクトの配列として指定します。各オブジェクトでは、name プロパティと value プロパティを指定する必要があります。

要求の例

- 要求のメソッド :

POST

- 要求の URL :

`https://localhost/api/deepsee/v1/Data/PivotExecute`

URL の有効な形式に関する説明は、"[Business Intelligence REST API の概要](#)" を参照してください。

- 要求の本文 :

```
{ "PIVOT": "Use in Dashboards/Product Info" }
```

別の例を示します。

```
{ "PIVOT": "Pivot Variables/Commission Calculator",
  "VARIABLES": [ { "name": "commissionpercentage", "value": 15 } ] }
```

応答の例

このサービスは、[POST /Data/MDXExecute](#) と同じ応答オブジェクトを返します。

POST /Info/Cubes

この REST 呼び出しを介してアクセスする InterSystems IRIS® データ・プラットフォーム・ネームスペースで使用できるキューブとサブジェクト領域に関する情報を返します。

URL パラメータ

なし。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

TYPE	オプション。このプロパティには、cubes または subjectareas を指定できます。cubes を指定した場合は、サーバはキューブに関する情報のみを送信します。subjectareas を指定した場合は、サーバはサブジェクト領域に関する情報のみを送信します。このプロパティを指定しない場合は、サーバはキューブとサブジェクト領域の両方に関する情報を送信します。
BASECUBE	オプション。指定されている場合、このプロパティの値はキューブの論理名と等しくする必要があります。この場合は、サーバはこのキューブに基づいたサブジェクト領域およびキューブに関する情報のみを送信します。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

https://localhost/api/deepsee/v1/Info/Cubes

URL の有効な形式に関する説明は、["Business Intelligence REST API の概要"](#) を参照してください。

- 要求の本文：

```
{ "BASECUBE": "patients" }
```

応答の例

```
{
  "Info": {
    "Error": "", "Type": "all", "BaseCube": "patients"
  },
  "Result": {
    "Cubes": [
      {
        "name": "AsthmaPatients", "displayName": "AsthmaPatients", "modTime": "2016-11-14
20:49:14", "type": "subjectArea"
      },
      {
        "name": "DemoMDX", "displayName": "DemoMDX", "modTime": "2016-11-14
20:49:14", "type": "subjectArea"
      },
      {
        "name": "YoungPatients", "displayName": "YoungPatients", "modTime": "2016-11-14
20:49:14", "type": "subjectArea"
      }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに Cubes プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、キューブまたはサブジェクト領域ごとに 1 つのオブジェクトがあります。これらのオブジェクト内では、type プロパティはアイテムがキューブなのかサブジェクト領域なのかを示します。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Info/Dashboards

この REST 呼び出しを介してアクセスする InterSystems IRIS ネームスペースで使用できるダッシュボードに関する情報を返します。

URL パラメータ

なし。

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- ・ 要求のメソッド :

POST

- ・ 要求の URL :

`https://localhost/api/deepsee/v1/Info/Dashboards`

URL の有効な形式に関する説明は、[“Business Intelligence REST API の概要”](#) を参照してください。

応答の例

```
{
  "Info": {
    "Error": ""
  },
  "Result": {
    "Dashboards": [
      {
        "fullName": "Basic Dashboard Demo",
        "name": "Basic Dashboard Demo",
        "lastModified": "2016-11-14 19:39:14",
        "itemType": "dashboard"
      },
      {
        "fullName": "Custom Drilldown Spec",
        "name": "Custom Drilldown Spec",
        "lastModified": "2016-11-14 19:39:14",
        "itemType": "dashboard"
      }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに Dashboards プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、ダッシュボードごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、[“応答オブジェクトに関する注意事項”](#) を参照してください。

POST /Info/FilterMembers/:datasource/:filterSpec

指定されたデータ・ソース (キューブまたは KPI) で定義されたように、指定されたフィルタのメンバに関する情報を返します。

URL パラメータ

datasource	<p>必須項目。データ・ソースの名前。これは、以下のいずれかです。</p> <ul style="list-style-type: none"> cubename — 論理キューブ名。 cubename.cube — 論理キューブ名とそれに続く .cube kpiname.kpi — 論理 KPI 名とそれに続く .kpi <p>この名前にはスラッシュを使用できます。“キューブ名と KPI 名でのスラッシュの使用”を参照してください。</p>
filterSpec	必須項目。フィルタ仕様。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。これらの両方のプロパティで、データ・ソースに適用されるフィルタを指定することで、このサービスによって返されるメンバのリストを絞り込みます。

RELATED	オプション。指定されている場合、このプロパティはオブジェクトの配列で、それぞれのオブジェクトには spec プロパティ (フィルタ仕様) と value プロパティ (そのフィルタの値) が格納されます。value プロパティは MDX セット式である必要があり、メンバ・キーを使用する必要があります。
SEARCHKEY	オプション。

要求の例

- 要求のメソッド :

POST

- 要求の URL :

https://localhost/api/deepsee/v1/Info/FilterMembers/:demomdx.kpi/:homed.h1.zip

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文 :

```
{
  "RELATED": [
    { "spec": "gend.h1.gender", "value": "&[female]" }
  ],
  "SEARCHKEY": "Jan"
}
```

応答の例

```
{
  "Info": {
    "Error": "", "DataSource": "demomdx.cube", "DataSourceType": "cube", "Default": "", "Filter": "[HomeD].[H1].[ZIP]"
  },
  "Result": {
    "FilterMembers": [
      { "text": "32006", "value": "&[32006]", "description": "" },
      { "text": "32007", "value": "&[32007]", "description": "" },
      { "text": "34577", "value": "&[34577]", "description": "" },
      { "text": "36711", "value": "&[36711]", "description": "" },
      { "text": "38928", "value": "&[38928]", "description": "" }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに FilterMembers プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、フィルタ・メンバごとに 1 つのオブジェクトがあります。指定されたフィルタ・メンバのオブジェクトには、以下のプロパティが含まれています。

- ・ description には、そのメンバのテキスト説明が含まれています (ある場合)。
- ・ text には、そのメンバの表示テキストが含まれています。
- ・ value には、そのメンバの論理値が含まれています (通常は MDX キー)。

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#) を参照してください。

POST /Info/Filters/:datasource

指定されたデータ・ソース (キューブまたは KPI) で使用できるフィルタに関する情報を返します。

URL パラメータ

datasource	<p>必須項目。データ・ソースの名前。これは、以下のいずれかです。</p> <ul style="list-style-type: none"> cubename — 論理キューブ名。 cubename.cube — 論理キューブ名とそれに続く .cube kpiname.kpi — 論理 KPI 名とそれに続く .kpi <p>この名前にはスラッシュを使用できます。“キューブ名と KPI 名でのスラッシュの使用”を参照してください。</p>
------------	--

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド :

POST

- 要求の URL :

`https://localhost/api/deepsee/v1/Info/Filters/:aviationevents`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

応答の例

```
{
  "Info": {
    "Error": "", "DataSource": "aviationevents.cube", "DataSourceType": "cube"
  },
  "Result": {
    "Filters": [
      { "caption": "Year", "value": "[EventDateD].[H1].[Year]", "type": "year" },
      { "caption": "Month", "value": "[EventDateD].[H1].[Month]", "type": "" },
      { "caption": "Day", "value": "[EventDateD].[H1].[Day]", "type": "" }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに Filters プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、フィルタごとに 1 つのオブジェクトがあります。各オブジェクトには以下のプロパティがあります。

- caption には、そのフィルタの表示値が含まれています。
- type には、フィルタ・タイプが含まれています (存在する場合)。このプロパティには、“calc” または時間関数の名前を指定できます。他の場合は、このプロパティは空です。
- value には、そのフィルタの論理識別子であるフィルタ仕様が含まれています。フィルタ仕様の詳細は、“[POST /Info/FilterMembers/:datasource/:filterSpec](#)”を参照してください。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Info/KPIs

この REST 呼び出しを介してアクセスする InterSystems IRIS ネームスペースで使用できる KPI に関する情報を返します。

URL パラメータ

なし。

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Info/KPIs`

URL の有効な形式に関する説明は、[“Business Intelligence REST API の概要”](#) を参照してください。

応答の例

```
{
  "Info": {
    "Error": ""
  },
  "Result": {
    "KPIs": [
      {
        "name": "Aviation Actions", "caption": "Aviation Actions", "moddate": "2016-11-14
11:22:08", "type": "kpi"
      },
      {
        "name": "AviationTopConcepts", "caption": "AviationTopConcepts", "moddate": "2016-11-14
11:22:08", "type": "kpi"
      },
      {
        "name": "BPDiaStolic", "caption": "BPDiaStolic", "moddate": "2016-11-14
11:22:08", "type": "kpi"
      },
      ...
    ]
  }
}
```

この応答オブジェクト内では、`Result` プロパティに `KPIs` プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、KPI ごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、[“応答オブジェクトに関する注意事項”](#) を参照してください。

POST /Info/ListingFields/:cube

この REST 呼び出しを介してアクセスする InterSystems IRIS ネームスペースで使用できる <listingField> 要素に関する情報を返します。

URL パラメータ

cube	必須項目。キューブの論理名。この名前にはスラッシュを使用できます。“ キューブ名と KPI 名でのスラッシュの使用 ”を参照してください。
------	---

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド :

POST

- 要求の URL :

`https://localhost/api/deepsee/v1/Info/ListingFields/:holefoods`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

応答の例

```
{
  "Info": {
    "Error": "", "BaseCube": "holefoods"
  },
  "Result": {
    "ListingFields": [
      { "caption": "Channel", "expression": "%EXTERNAL(Channel) Channel" },
      { "caption": "City", "expression": "Outlet->City" },
      { "caption": "Comment", "expression": "Comment" },
      ... ]
  }
}
```

この応答オブジェクト内では、Result プロパティに ListingFields プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、<listingField> 要素ごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Info/Listings/:cube

指定されたキューブで利用できる詳細リストに関する情報を返します。

URL パラメータ

cube	必須項目。キューブの論理名。この名前にはスラッシュを使用できます。“ キューブ名と KPI 名でのスラッシュの使用 ”を参照してください。
------	---

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

TYPE	オプション。このプロパティには、map または table を指定できます。このプロパティが map の場合は、サーバはマップ・リストに関する情報のみを送信します。このプロパティが table の場合は、サーバは非マップ・リストに関する情報のみを送信します。このプロパティを指定しない場合は、サーバはすべての種類のリストに関する情報を送信します。
------	---

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Info/Listings/:holefoods`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

- 要求の本文：

```
{ "TYPE": "map" }
```

応答の例

別の例を示します。

```
{
  "Info": {
    "Error": "", "DataSource": "holefoods", "Type": "all"
  },
  "Result": {
    "Listings": [
      {
        "name": "Custom Listing",
        "name": "Another Sample Listing by Date",
        "fields": "%ID As \"ID #\", DateOfSale As \"Sale Date\"",
        "order": "DateOfSale, %ID",
        "type": "table",
        "source": "listingGroup",
        "edit": 1
      },
      {
        "name": "Another Sample Listing with Customer Info",
        "fields": "%ID, Outlet->City \"Store Location\", Outlet->Country->Name Country, Product->Name Product, ZipCode \"Customer ZipCode\", Latitude, Longitude", "order": "",
        "type": "map",
        "source": "listingGroup",
        "edit": 1
      },
      {
        "name": "Customer Info",
        "fields": "%ID, Outlet->City \"Store Location\", Outlet->Country->Name Country, Product->Name Product, ZipCode \"Customer ZipCode\", Latitude, Longitude", "order": "",
        "type": "map",
        "source": "cube",
        "edit": 0
      }
    ]
  }
}
```



```
    ...]
  }
}
```

この応答オブジェクト内では、`Result` プロパティに `Listings` プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、詳細リストごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、[“応答オブジェクトに関する注意事項”](#) を参照してください。

POST /Info/Measures/:cube

指定されたキューブで利用できるメジャーに関する情報を返します。

URL パラメータ

cube	必須項目。キューブの論理名。この名前にはスラッシュを使用できます。“ キューブ名とKPI名でのスラッシュの使用 ”を参照してください。
------	---

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

SKIPCALCULATED	オプション。
----------------	--------

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Info/Measures/:demomdx`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

応答の例

```
{
  "Info": {
    "Error": "",
    "BaseCube": "DemoMDX",
    "SkipCalculated": 0
  },
  "Result": {
    "Measures": [
      { "name": "%COUNT", "caption": "%COUNT", "type": "integer", "hidden": 0, "factName": "" },
      { "name": "Age", "caption": "Age", "type": "integer", "hidden": 0, "factName": "MxAge" }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに Measures プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、メジャーごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Info/NamedFilters/:cube

指定されたキューブで使用できる名前付きフィルタに関する情報を返します。

URL パラメータ

cube	必須項目。キューブの論理名。この名前にはスラッシュを使用できます。“ キューブ名と KPI 名でのスラッシュの使用 ”を参照してください。
------	---

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Info/NamedFilters/:holefoods`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

応答の例

```
{
  "Info": {
    "Error": "", "BaseCube": "holefoods"
  },
  "Result": {
    "NamedFilters": [
      {
        "name": "Sample Named Filter",
        "description": "",
        "spec": "[Product].[P1].[Product Category].&[Seafood]", "cube": "HOLEFOODS"
      }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに NamedFilters プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、名前付きフィルタごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Info/Pivots

この REST 呼び出しを介してアクセスする InterSystems IRIS ネームスペースで使用できるピボット・テーブルに関する情報を返します。

URL パラメータ

なし。

要求の本文の詳細

このサービスでは、要求本文の以下のプロパティが使用されます。

BASECUBE	オプション。指定されている場合、このプロパティの値はキューブの論理名と等しくする必要があります。この場合は、サーバはこのキューブに基づいたピボット・テーブルに関する情報のみを送信します。
----------	---

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Info/Pivots`

URL の有効な形式に関する説明は、["Business Intelligence REST API の概要"](#) を参照してください。

- 要求の本文：

```
{ "BASECUBE": "PATIENTS" }
```

応答の例

```
{
  "Info": {
    "Error": "", "BaseCube": ""
  },
  "Result": {
    "Pivots": [
      {
        "fullName": "Calculated Members\\Alternative Avg Allergy Count",
        "name": "Alternative Avg Allergy Count",
        "lastModified": "2016-11-14 11:22:08",
        "itemType": "pivot"
      },
      {
        "fullName": "Calculated Members\\Average Patient Count per Decade",
        "name": "Average Patient Count per Decade",
        "lastModified": "2016-11-14 11:22:08",
        "itemType": "pivot"
      }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに Pivots プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、ピボット・テーブルごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、["応答オブジェクトに関する注意事項"](#) を参照してください。

POST /Info/PivotVariableDetails/:cube/:variable

指定されたピボット変数に関する詳細情報を返します。

URL パラメータ

cube	必須項目。指定されたピボット変数にアクセスできるキューブの論理名。この名前にはスラッシュを使用できます。“ キューブ名と KPI 名でのスラッシュの使用 ”を参照してください。
variable	必須項目。ピボット変数の論理名。

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Info/PivotVariableDetails/:holefoods/:Year`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

応答の例

```
{
  "Info": {
    "Error": "", "BaseCube": "holefoods", "Variable": "Year"
  },
  "Result": {
    "Details": {
      "context": "literal", "defaultValue": "NOW", "description": "", "displayList": "", "displayName": "Year", "name": "Year", "sourceName": "HoleFoods.kpi", "sourceType": "kpi", "type": "string", "valueList": ""
    }
  }
}
```

この応答オブジェクト内では、Result プロパティに Details プロパティが含まれています。このプロパティにはピボット変数の詳細情報が含まれています。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Info/PivotVariables/:cube

指定されたキューブで使用できるピボット変数に関する情報を返します。

URL パラメータ

cube	必須項目。キューブの論理名。この名前にはスラッシュを使用できます。“ キューブ名と KPI 名でのスラッシュの使用 ”を参照してください。
------	---

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Info/PivotVariables/:holefoods`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

応答の例

```
{
  "Info": {
    "Error": "", "BaseCube": "holefoods"
  },
  "Result": {
    "PivotVariables": [
      {
        "name": "CommissionPercentage", "caption": "Commission
Percentage", "defValue": "0", "context": "literal", "desc": ""
      },
      {
        "name": "Year", "caption": "Year", "defValue": "NOW", "context": "literal", "desc": ""
      }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに PivotVariables プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、ピボット変数ごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

POST /Info/QualityMeasures/:cube

指定されたキューブで利用できる品質メジャーに関する情報を返します。

URL パラメータ

cube	必須項目。キューブの論理名。この名前にはスラッシュを使用できます。“ キューブ名と KPI 名でのスラッシュの使用 ”を参照してください。
------	---

要求の本文の詳細

このサービスでは要求の本文は無視されます。

要求の例

- 要求のメソッド：

POST

- 要求の URL：

`https://localhost/api/deepsee/v1/Info/QualityMeasures/:holefoods`

URL の有効な形式に関する説明は、“[Business Intelligence REST API の概要](#)”を参照してください。

応答の例

```
{
  "Info": {
    "Error": "", "BaseCube": "holefoods"
  },
  "Result": {
    "QualityMeasures": [
      { "name": "TestCatalog\\TestSet\\TestQM", "caption": "Sample Quality Measure", "description": "" }
    ]
  }
}
```

この応答オブジェクト内では、Result プロパティに QualityMeasures プロパティが含まれています。このプロパティにはオブジェクトの配列が含まれていて、品質メジャーごとに 1 つのオブジェクトがあります。

すべての応答オブジェクトに適用される情報は、“[応答オブジェクトに関する注意事項](#)”を参照してください。

