Blueprint Analytics

User Guide

2019-05-22

About Blueprint Analytics

Blueprint Analytics reporting is a dynamic mode of modeling raw data into powerful resources for stakeholders. Loading Blueprint project data, you can use the PowerPivot add-in within Microsoft Excel to generate reports that help you discover business solutions and improve matrices for decision making.

PowerPivot allows you to model data in the following formats:

- PivotCharts
- PivotTables

Purpose

The intended audience of this document is intermediate users that have experience producing reports for business purposes. Users of this document should be familiar with Microsoft Excel and the basics of data modeling. This document aims to draw upon the user's reporting skills and Blueprint repository data to foster an expansion of competitive intelligence.

Previous experience with Blueprint is necessary to produce Blueprint Analytics reports. It is highly recommended that you have a solid understanding of the Blueprint artifact properties you intend to load in PowerPivot. You must also have access to the Blueprint projects you want to include in your reports.

Additional resources

The Microsoft Business Intelligence web site contains a comprehensive database of PowerPivot help topics.

The following articles from the Microsoft Business Intelligence web site outline how to use PowerPivot:

- Create a PivotChart
- Create a PivotTable to analyze worksheet data
- Filter the data you import into PowerPivot
- Calculated Fields in PowerPivot
- Create relationships in Diagram View in PowerPivot

blueprint®

Overview

Blueprint is typically used to store key information about an enterprise's IT projects. Employing a data connector and a business intelligence tool set, Blueprint Analytics allows you to explore key project information early on, possibly answering important questions about the condition of your project.

Blueprint Analytics provides a reporting layer on top of the Blueprint repository that exposes data via standard Open Data Protocol (OData). Analyzing the data in your Blueprint database, Blueprint Analytics produces reports that provide business insight into the state of your projects, both for you and stakeholders. By means of filtering and data modeling, this mode of reporting gives you full control and flexibility with your data, optimizing visual and diagrammatic communication in the process.

Licensed database users can produce Blueprint Analytics reports.

To help users get started with modeling data, this brief guide outlines software requirements, high-level steps, instructions on how to load Blueprint data and conversion information.

You can generate a Blueprint Analytics report by completing the following high level steps:

1. **Ensure your environment meets the system requirements**
2. **Import Blueprint data into PowerPivot**
    - **Familiarize yourself with the artifact list and entities**
3. **Filter and model your data**

System Requirements

Blueprint users that want to produce Blueprint Analytics reports require the following:

- Microsoft Excel 2010 or 2013
- PowerPivot add-in for Microsoft Excel

  > **Note:** Users with Microsoft Excel 2010 must download and install PowerPivot separately. For downloading and installation instructions, see *Download PowerPivot for Excel 2010*.
  > Microsoft Excel 2013 comes with the PowerPivot add-in installed. For more information on accessing the PowerPivot add-in within Excel 2013, see *Start PowerPivot in Microsoft Excel 2013 add-in*.

- A Blueprint Analytics license

  Contact Blueprint Support through the Customer Portal (http://portal.blueprintsys.com).

- A Blueprint database user account with instance-level or project-level role permissions
- A Blueprint Analytics URL, including the port number

  For example: **`http://yourcompany.blueprint.com:8081`**

Acquiring the Right Level of Access

In order to create reports in PowerPivot using the relevant project data, your Blueprint database user account requires the proper permissions to access the Blueprint data.

Depending on whether you want to load data from a certain project, many projects or the entire instance, make sure you have a database user account with the applicable level of privileges. The following options exist:

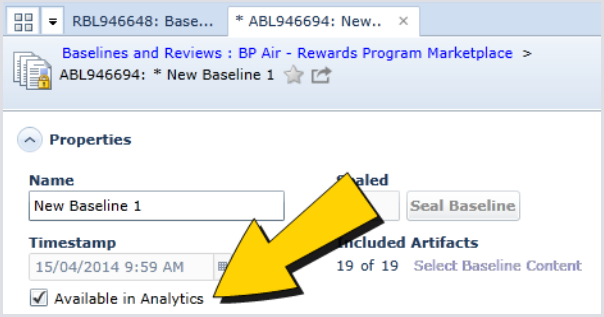
- Privileges at the Instance Administration Role level

  This option gives you access to all project data. Although Instance Administration privileges provide an efficient way of getting access to multiple projects, loading all Blueprint instance data may cause import performance to become slower.

- **Can Report** privileges at the Project Role Assignment level

  This option works best when you only need to work with data from one project or a low number of projects. This option is a more complicated process than granting Instance Administration privileges. You need to repeat this process in the Project Administration Console for every project that needs to be accessed.

Note: In addition to acquiring the right permissions at the instance-level or project-level, the **Available in Analytics** check box must be enabled in each individual baseline within Blueprint that you want to include in your report.



The following sections detail how to create and assign instance-level and project-level permissions to use Blueprint Analytics.

Acquiring Instance-Level Privileges

This option gives you access to all project data. Although Instance Administration privileges provide an efficient way of getting access to multiple projects, loading all Blueprint instance data may cause import performance to become slower.

The steps involved in obtaining instance-level privileges are as follows:

- Create a Blueprint Analytics role (if one does not already exist)
- Assign the Analytics role

Creating a Blueprint Analytics Role (Instance-Level)

To add the role:

1. Open the *Instance Admin Console*.
2. Click **Instance Administrator Roles** (*Manage Administrator Roles* group).
   The *Instance Administrator Roles* page appears.
3. Click the **New** icon (*Manage Items* group).
4. In the *Instance Administrator Role* pane, enter **Blueprint Analytics**.

   Enter a description in the **Description** field so you know what privileges you are selecting when you assign the instance administrator role. For example: *User can produce Blueprint Analytics reports using OData on all projects.*
5. In the *Role Privileges* list, select the Analytics privilege check box.
6. Click **Save**.

The new role appears in the list of Instance Administrator Roles.

Next, assign the new role to a user. See Assigning an Instance Administrator role to a user.

Assigning the Analytics Role (Instance-Level)

(missing or bad snippet)

Acquiring Project-Level Privileges

This option works best when you only need to work with data from one project or a low number of projects. This option is a more complicated process than granting Instance Administration privileges. You need to repeat this process in the Project Administration Console for every project that needs to be accessed.

The steps involved in obtaining project-level privileges are as follows:

- Create a Blueprint Analytics role (if one does not already exist)
- Assign the Blueprint Analytics role

Creating a Blueprint Analytics Role (Project-Level)

This role must be re-created for every project that needs to be accessed via Blueprint Analytics.

To create the role:

1. Open the *Project Roles* tab.
   1. Open the *Project Admin Console*.
   2. Click the **Project Roles** link.
      The *Project Roles* tab is displayed.
2. Click the **New** button.

A new project role appears in the Project Role list.

3. Enter *Blueprint Analytics* in the **Name** field.
4. Provide a description for the project role.
5. Select the **Can Report** check box.
6. Select the project administrator role with the administrative privileges that you want the project role to have.
7. Click **Save**.

The new role appears in the list of Project Roles.

Next, assign the new role to a user. See Assign the Blueprint Analytics Role (Project-Level).

Assign the Blueprint Analytics Role (Project-Level)

The Blueprint Analytics role must be assigned to the user in each project that needs to be accessed via Blueprint Analytics.

(missing or bad snippet)

Importing Blueprint data in PowerPivot

> **Note:** Before loading Blueprint data into Microsoft Excel, users require Microsoft Excel 2010 (or higher), PowerPivot and licensing. For more information, see System Requirements.

You need to load your Blueprint data into PowerPivot so the data can be accessed and analyzed.

To load Blueprint data into PowerPivot:

1. Open Microsoft Excel.
2. Click the **Manage** button (*Data Model* group, *PowerPivot* tab).
   - For Microsoft Excel 2010, click **PowerPivot Window** (*PowerPivot* tab).

   The *PowerPivot for Excel* window appears.
3. Click **From Data Service** (*Get External Data* group).
   - For Microsoft Excel 2010, click **From Data Feeds** (*Get External Data* group).
4. In the **Data Feed Url** field, enter the URL and port number of the Blueprint Analytics instance (for example: `http://yourcompany.blueprint.com:8081`). Add a forward slash to the end and attach one of following:
   - `ArtifactDataService.svc`

     Allows you to access any Blueprint data you have permissions to, *except* historical, review and baseline data.

     > **Note:** This data set potentially takes less time to load than the following data set.
   - `HistoricalArtifactDataService.svc`

     Allows you to access any Blueprint data you have permissions to, *including* historical, review and baseline data.

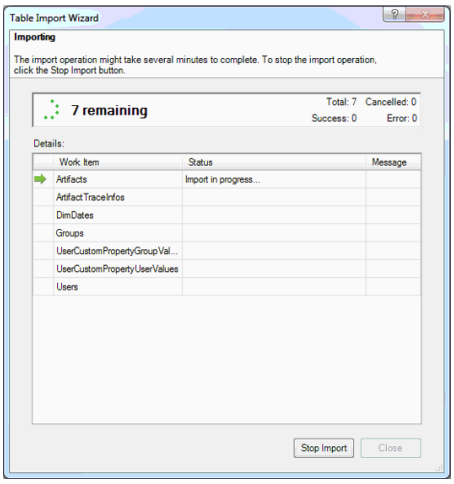
   Depending on the data set you use, your **Data Feed Url** should look similar to the following: `http://yourcompany.blueprint.com:8081/HistoricalArtifactDataService.svc/`
5. Click the **Advanced** button.

   The *Advanced* dialog appears.
6. Select **Basic** on the **Integrated Security** menu.
7. Enter your password.
8. Select **True** on the **Persist Security Info** menu.
9. Enter your Blueprint user name in the **User ID** field.
10. Enter the same data feed URL you entered in step #3 in the **Data Source** field.
11. Enter the same data feed URL in the **Service Document Url** field.
12. Click the **Test Connection** button.
13. Click **OK**.

    The *Select Table and Views* screen appears.
14. Select the check box at the top left to select all of the check boxes.

15. Click **Finish**.

    The *Importing* screen appears.



> **Note**: Please be patient. The import process can take an extended period of time, depending on the size of your projects. If there is a problem with the import, an error message is displayed.

16. If the import is successful, the *Success* dialog appears.

    Click **Close**.

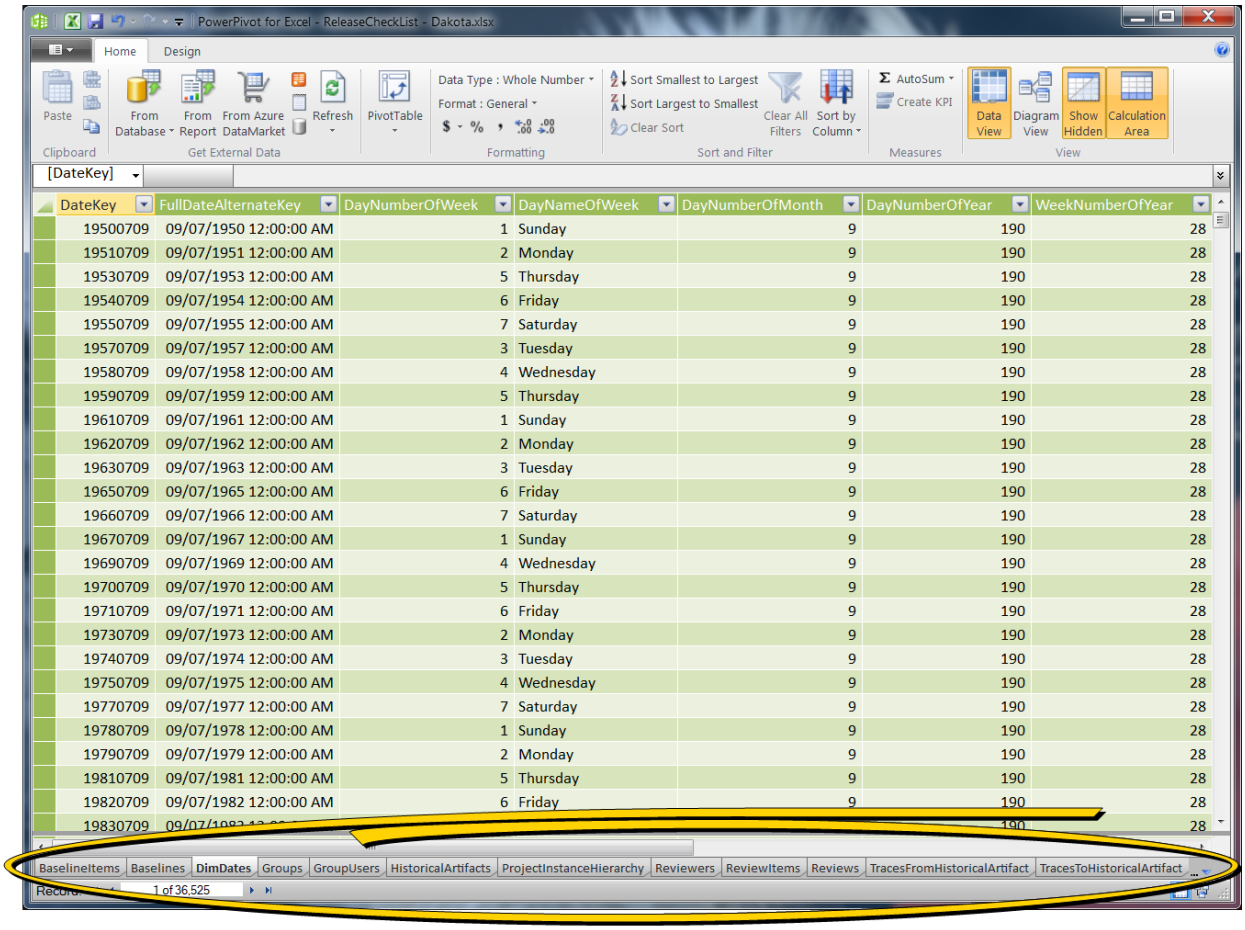
Your Blueprint data appears within PowerPivot cells.

(missing or bad snippet)

Viewing and Understanding Your Blueprint Data in PowerPivot

Note: Viewing textual properties, such as **Description**, is not supported in PowerPivot.

Blueprint data is converted in PowerPivot using OData protocol. In *Data View*, your Blueprint data displays in spreadsheet format and is arranged into entity sets that are accessible at the bottom of the window. Each entity set or tab contains entities that are arranged into columns on the worksheet. For more information on entities and a list of entities, see About entities.



(missing or bad snippet)

The PowerPivot layout versus the Blueprint artifact hierarchy

The layout of entities in PowerPivot columns loosely resembles the original artifact hierarchy in Blueprint. For example, the properties that appear at the top of the artifact hierarchy in Blueprint (such as **Name** or **Artifact ID**) also appear in the entity column titles in PowerPivot.

However, the Blueprint parent-child artifact hierarchy is *flattened* in PowerPivot. Additionally, the OData protocol specially encodes specific Blueprint properties and displays them in different ways. See the following sections for more information on property encoding.

Custom properties

Custom properties with unique names do not have any special encoding in PowerPivot unless they contain special characters. However, properties with names that are shared across projects are numerically encoded by OData protocol. For example: **Risk1**, **Risk2**, and so on.

Properties containing numbers and/or special characters

All numbers and special characters in property names and property values are encoded using OData standards. For example: if a property value starts with a number or an underscore, the property value appears as **C_[propertyvalue]**.

> **Example**
>
> *323* becomes *C_323*

Similar properties across projects

When a property name and a property type are identical within two or more loaded projects, all associated values are merged into a single column.

Properties with identical names and *different* property types (for example, a single choice type versus a multiple choice type) appear in their own separate columns. The columns are arranged according to numerical order.

Multiple choice properties

Multiple choice properties are flattened in PowerPivot. Each multiple choice value has its own column and, in the rows below, a boolean value appears depending on whether the value has been selected in the original Blueprint project. **0** denotes a non-selected value and **1** denotes a selected value.

> **Example**
>
> Miguel, a business analyst, has a *BP Air* project in Blueprint that he wants to load in PowerPivot. In his Blueprint project, there is a multiple choice property called **ProductType** that contains the following values: **Desktop**, **Mobile** and **Service**. After loading his data into PowerPivot, three columns appear that are named **ProductType_Desktop**, **ProductType_Mobile** and **ProductType_**

Service. Within Blueprint a user had selected **Desktop** and **Mobile** so the two columns **ProductType_Desktop** and **ProductType_Mobile** have **1** (selected) and **ProductType_Service** has **0** (non-selected).

About entities

ItemId	ItemName	VersionNumber	ArtifactTypeId	ArtifactTypeName	PrimitiveItemTypePredefined	PrimitiveTypeName	CreatedOn
3	NGTV	145	57	Project	4097	Project	12/03/2012 3:20:2...
4	Baselines and Reviews	1	76	Base Baseline & Revi...	4353	BaselineFolder	12/03/2012 3:20:3...
6	Rule Repository Rep...	1	94	Folder	4102	PrimitiveFolder	12/03/2012 3:20:3...
7	Reference Documents	1	94	Folder	4102	PrimitiveFolder	12/03/2012 3:20:3...
12	Remote App	4	94	Folder	4102	PrimitiveFolder	12/03/2012 3:20:3...
13	MODEL INFORMATION	2	93	Textual Requirement	4101	TextualRequirement	12/03/2012 3:20:3...
14	Requirements	1	94	Folder	4102	PrimitiveFolder	12/03/2012 3:20:3...
15	Assumptions and Con...	1	94	Folder	4102	PrimitiveFolder	12/03/2012 3:20:3...
16	Ministry Technology ...	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
17	Current Legislation	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
18	Support a Public Syst...	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
19	Data Security Standar...	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
20	Promote a simplified ...	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
21	WIP Information is Pr...	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
22	Hours of Operation	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
23	Flexible Architecture...	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
24	Implement instantan...	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
25	Minimize deploymen...	2	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
26	Support Overlapping ...	3	158	Assumption and Con...	4101	TextualRequirement	12/03/2012 3:20:3...
27	Authentication Requi...	1	94	Folder	4102	PrimitiveFolder	12/03/2012 3:20:3...
28	System must require ...	1	93	Textual Requirement	4101	TextualRequirement	12/03/2012 3:20:3...
29	Docket Requirements	1	94	Folder	4102	PrimitiveFolder	12/03/2012 3:20:3...

An entity is an OData resource that contains and/or structures your Blueprint data. Some entities contain Blueprint properties (including custom properties, single choice properties, multiple choice properties and so on). Other entities are data-related keys (such as **ItemKey**, **CreatedOnDateKey** and so on) that you can link to other data in order to generate reports on trends.

Note: Each type of entity can be modeled into PivotCharts and PivotTables.

For an overview of entities and other OData resources, go here: http://www.odata.org/documentation/odata-version-3-0/odata-version-3-0-core-protocol/

Blueprint data is flattened into the following categories/entity sets:

- Artifacts
- ArtifactTraces
- Baselines
- BaselineItems
- DimDates
- Groups
- GroupUsers
- HistoricalArtifacts
- ProjectInstanceHierarchy
- Reviewers
- ReviewItems
- Reviews
- UserCustomPropertyGroupValues
- UserCustomPropertyUserValues
- Users

blueprint

Artifacts

- **ItemId**: The ID of the item in Blueprint (numerical value).
- **ItemName**: The name of the item.
- **VersionNumber**: The number of revisions the item has undergone and, as such, the current revision number.
- **ArtifactTypeName**: Identifies the name of the artifact type (descriptive string value).
- **PrimitiveItemTypePredefined**: Base artifact type, such as textual requirement (numerical value).
- **PrimitiveTypeName**: The type of the item. For example: project, primitivefolder (a basic folder), baselinefolder (a folder designed to contain baselines) and so on.
- **CreatedOn**: The date the item was created on.
- **CreatedOnDateKey**: The key entity you have to link to other similar items in *Diagram View* to optimize the filter output of this element (**Created On** property).
- **CreatedBy**: Numerical identification for the user ID that created the artifact.
- **CreatedByName**: The user name that created the item.
- **LastModifiedOn**: The date and time the item was last modified.
- **LastModifiedOnDateKey**: The key entity you have to link in order to see data related to the last modifications.
- **LastModifiedBy**: Numerical identification for the user ID that last modified the artifact.
- **LastModifiedByName**: The user that last modified the item.
- **ParentId**: The ID of the parent of the item.
- **AllParents**: Lists the parents of the item by ID.
- **ItemLevel**: Identifies the level in the hierarchy where the item resides (numerical value).
- **ParentLevel2**: Identifies the ID of parent at the second highest level in relation to the item. With **ParentLevel3**, the third highest item is identified and so on.
- **ParentLevel2Name**: Identifies the name of the second highest level parent in relation to the item. With **ParentLevel3Name**, identifies the name of the parent at the third highest level and so on.

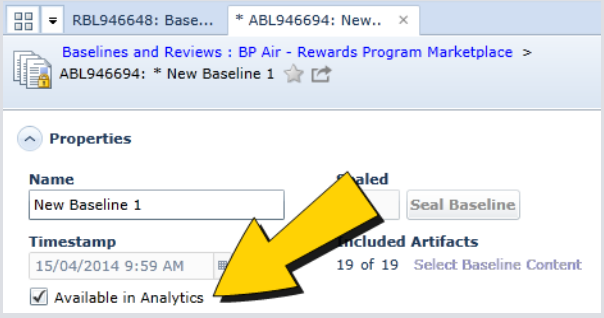
ArtifactTraces

- **ArtifactTraceId**: Numerical identifier of the trace ID.
- **SourceArtifactId**: The artifact ID from where the trace starts (that is, the first half of the trace relationship).
- **TargetArtifactId**: The second half of the trace relationship (artifact ID).
- **TraceTypeId**: Numeric identifier for the trace type ID.
- **TraceTypeName**: The type of trace, such as **Manual** or **Parent/Child**.
- **Suspect**: A boolean that indicates whether the trace is suspect or not.

Baselines

- **BaselineId**: Numerical identifier of the baseline.
- **BaselineStatus**: The status of the baseline. The baseline can be sealed or unsealed.
- **ItemName**: The name of the baseline artifact.
- **VersionNumber**: The number of revisions the item has undergone and, as such, the current revision number.
- **ArtifactTypeId**: Numerical identifier of the trace ID.
- **ArtifactTypeName**: Identifies the type of artifact.
- **PrimitiveTypeName**: The name of the primitive artifact type.
- **CreatedOn**: The date the item was created on.
- **CreatedOnDateKey**: Link this key to date entities in the **DimDates** set to facilitate reporting on the date dimension (*Diagram View*).
- **CreatedBy**: Numerical identification for the user ID that created the artifact.
- **CreatedByName**: The user name that created the item.
- **LastModifiedOn**: The date and time the item was last modified.
- **LastModifiedOnDateKey**: Link this key to date entities in the **DimDates** set to facilitate reporting on the date dimension (*Diagram View*).
- **LastModifiedBy**: Numerical identification for the user ID that last modified the artifact.
- **LastModifiedByName**: The user that last modified the item.
- **ProjectId**: The numerical identifier of the project.
- **ProjectName**: The name of the project.
- **ContentsForAnalytics**: Indicates if the contents of the baseline are available for Blueprint Analytics reporting.

Note: In addition to acquiring the right permissions at the instance-level or project-level, the **Available in Analytics** check box must be enabled in each individual baseline within Blueprint that you want to include in your report.



BaselineItems

- **BaselineId**: Numerical identifier of the baseline ID.
- **ItemKey**: Use this item key to link baseline items with artifacts in the *HistoricalArtifacts* entity set (*Diagram View*).

DimDates

This category contains entities that you can link to in order to report on the date dimension.

- **DateKey**: Use this key to link date items together in the *Diagram View*. **DateKey** can show you time trends. Creates aggregations for the year or month of your choosing. For example: **20130709** signifies July 9, 2013
- **FullDateAlternateKey**: An alternate key with date and time information you can use to link date items together.
- **DayNumberOfWeek**: Numerically represents the day of the week.
- **DayNameOfWeek**: The name of the day of the week.
- **DayNumberOfMonth**: The day number in the month.
- **DayNumberOfYear**:The day number within the year.
- **WeekNumberOfYear**: The week number within the year.
- **MonthName**: The name of the month.
- **MonthNumberOfYear**: The number of the month within the year.
- **CalendarQuarter**: The calendar quarter number.
- **CalendarYear**: The year.
- **CalendarSemester**: The numerical calendar semester.

Groups

- **GroupId**: The numerical identifier of group ID in Blueprint.
- **GroupName**: The name of group in Blueprint.
- **ParentGroupId**: The numerical identifier of the parent to this group, if it has one (could be null)
- **Email**: The email address of the group, if it was provided.
- **IsDeleted**: A boolean that indicates whether the item no longer exists.

GroupUsers

The link entity that you can use to link groups with users.

- **Groups_GroupId**: The group ID (numerical).
- **Users_UserId**: The user ID (numerical).

blueprint

HistoricalArtifacts

Note: There are limits to the amount of historical data you can load in each instance. Up to 350 unique custom properties can be loaded. Additionally, you can load up to 100 unique numerical, date-related or user-related properties (that is, 100 from each of the listed categories).

- **ItemKey**: The primary key for the artifact within the baseline. Use this column to link the baseline with the artifact.
- **ItemId**: The ID of the item in Blueprint (numerical identifier).
- **ItemName**: The name of the historical artifact.
- **VersionNumber**: The number of revisions the item has undergone and, as such, the current revision number.
- **ArtifactTypeId**: Numerical identifier of the trace ID.
- **ArtifactTypeName**: Identifies the type of artifact.
- **PrimitiveTypeName**: Identifies the primitive type of the artifact.
- **CreatedOn**: Identifies the date the artifact was created.
- **CreatedOnDateKey**: Link this key to date entities in the **DimDates** set to facilitate reporting on the date dimension (*Diagram View*). This is the integer representation of the date the review was created on.
- **CreatedBy**: Numerical identification for the user ID that created the artifact.
- **CreatedByName**: The user that last modified the item.
- **LastModifiedOn**: The date and time the item was last modified.
- **LastModifiedOnDateKey**: The key to link to date entities in the **DimDates** set to facilitate reporting on the date dimension (*Diagram View*).
- **LastModifiedBy**: Numerical identification for the user ID that last modified the artifact.
- **LastModifiedByName**: The user that last modified the item.
- **ProjectItemId**: The ID of the project item in Blueprint.
- **AllParents**: Lists the parents of the item by ID.
- **ItemLevel**: Identifies the level in the hierarchy where the item resides (numerical value).
- **ParentKey**: Link this key to other similar items in *Diagram View* to optimize parent filter output.
- **ParentItemId**: The ID of the parent item.
- **RevisionId**: The ID of the revision.
- **ProjectKey**: Link this key to other similar items in *Diagram View* to optimize the output of project information.
- **ParentLevel2**: Identifies the ID of the parent at the second highest level in relation to the item. With **ParentLevel3**, the third highest item is identified and so on.
- **ProjectName**: Identifies the name of the project in Blueprint.
- **ParentLevel2Name**: Identifies the name of the second highest level parent in relation to the item. With **ParentLevel3Name**, identifies the name of the parent at the third highest level and so on.

blueprint

ProjectInstanceHierarchy

Describes the instance and hierarchy containing the project.

- **ProjectId**: The project ID (numerical).
- **ParentFolderPath**: The hierarchy path string name.
- **ParentFolderLevel**: Identifies which level the folder is in within the hierarchy (1, 2, 3…).
- **ParentFolder1Name**: The name of the parent folder.

Reviewers

- **UserId**: The user ID (numerical).
- **ReviewId**: The review ID (numerical).
- **ItemKey**: Use this key to link baseline items together in the *Diagram View*. **ItemKey** can show you baseline trends and creates aggregations for the baseline of your choice.
- **UserName**: Identifies the user by name.
- **CommentCount**: Identifies the number of comments.
- **Permission**: A boolean that identifies whether the review participant is an approver or a reviewer.
- **ESignedOn**: If the review author requested an electronic signature, this entity identifies the date and time the electronic signature was given.
- **ESignedOnDateKey**: The item key you use to link electronic signature date items together in the *Diagram View*. **ESignedOnDateKey** can show you electronic signature date trends. Creates aggregations for the electronic signature of your choice.
- **State**: Identifies the state of the electronic signature, such as **approved** or **disapproved**, among others.

ReviewItems

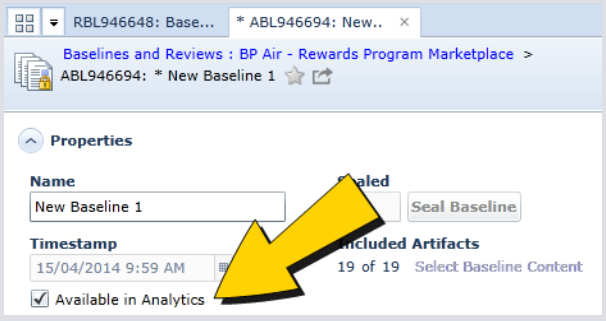
- **ReviewId**: The review ID (numerical).
- **ItemKey**: Use this key to link baselines with artifacts in the *HistoricalArtifacts* entity (*Diagram View*).
- **BaselineId**: The ID of the baseline (numerical).
- **Approval**: A boolean that identifies whether approval has been requested for this item in the review. Possible values are **requested** and **not requested**.

Reviews

- **ReviewId**: The review ID (numerical).
- **ReviewName**: Identifies the name of the review.
- **ReviewStatus**: Identifies the status of the review. For example: the review can have a draft or active status, among others.

- **ReviewType**: A boolean that identifies the type of review. Possible values are **informal** and **formal**.
- **BaselineId**: The baseline ID (numerical).
- **ReviewStartDate**: Identifies the date and time the review was started.
- **ReviewStartDateKey**: The integer representation of the review start date that facilitates reporting on the date dimension. This key can help you report on time trends.
- **ReviewEndDate**: Identifies the date and time the review was ended.
- **ReviewEndDateKey**: Use this key to link date items together in the *Diagram View*.
- **ReviewClosedDate**: Identifies the time and date the review has been closed.
- **ReviewClosedDateKey**: This key can show you time trends in *Diagram View*.
- **VersionNumber**: The number of revisions the item has undergone and, as such, the current revision number.
- **CreatedOn**: The date the item was created on.
- **CreatedOnDateKey**: Use this key to link to other similar items in *Diagram View* to optimize the filter output of this element.
- **CreatedBy**: Numerical identification for the user ID that created the artifact.
- **CreatedByName**: The user name that created the item.
- **LastModifiedOn**: The date and time the item was last modified.
- **LastModifiedOnDateKey**: The integer representation for the last modified date to help facilitate reporting on date dimension. Use this key to see data related to last modifications.
- **LastModifiedBy**: Numerically identifies the user that last modified the item.
- **LastModifiedByName**: Identifies the name of the user that last modified the item.
- **ProjectId**: The numerical identifier of the project.
- **ProjectName**: The name of the project.
- **ESignature**: A boolean that identifies whether the review author requested electronic signatures from review participants. Possible values are **requested** and **not requested**.
- **ContentsForAnalytics**: A boolean that indicates whether the contents of the artifact are available for Blueprint Analytics reporting.

Note: In addition to acquiring the right permissions at the instance-level or project-level, the **Available in Analytics** check box must be enabled in each individual baseline within Blueprint that you want to include in your report.



UserCustomPropertyGroupValues

- **ItemId**: The item ID.
- **GroupId**: The group ID (numerical).
- **PropertyName**: The name of the property.

UserCustomPropertyUserValues

- **ItemId**: The item ID.
- **GroupId**: The group ID (numerical).
- **PropertyName**: The name of the property.

Users

- **UserId**: The user ID (numerical).
- **UserName**: The user name.
- **DisplayName**: The display name of the user, if provided.
- **FirstName**: The first name of the user, if provided.
- **LastName**: The last name of the user, if provided.

blueprint

Filter and model your data

The following Microsoft Business Intelligence articles address data modeling, modifying scope and arranging data relationships:

- Create a PivotChart
- Create a PivotTable to analyze worksheet data
- Filter the data you import into PowerPivot
- Calculated Fields in PowerPivot