Deltek

Deltek Acumen 8.11

Application Programming Interface (API) Guide

December 5, 2025

Deltek

While Deltek has attempted to verify that the information in this document is accurate and complete, some typographical or technical errors may exist. The recipient of this document is solely responsible for all decisions relating to or use of the information provided herein.

The information contained in this publication is effective as of the publication date below and is subject to change without notice.

This edition published December 2025.

Deltek

Contents

Overview 1
Architecture of the Acumen API 2
Workflow 2
API Configuration File 3
Add the API Configuration File Location to Acumen 3
API Configuration File Requirements 3
Editing an API Configuration File 4
Example of an API Configuration File 4
API Data Model 6
Workbook Module Class Diagram 6
Workbook 6
Costs 7
Durations 7
Activity 8
Field 10
Relationship 11
Project 11
Snapshot 12
Logic Analyzer 12
Ribbon View Module Class Diagram 14
Ribbon View 14
Phase 14
Ribbon 15
Intersection 15
Ribbon View Metric 16
Metric Value 16
Metric Background Color 17
Metric Library Module Class Diagram 18
Metric Library 18
Metric Group 18
Metric 19
Threshold 20

Metric Filter 20
Metric Filter Expression 21
Forensic Report Module Class Diagram 22
Forensic Report 22
Projects Tab 23
Project Forensic Check 23
Project Forensic Check Comparison 23
Activities Tab 23
Activity Forensic Check 24
Activity Forensic Check Comparison 24
Relationships Tab 24
Relationships Forensic Check 24
Relationship Forensic Check Comparison 25
Resources Tab 25
Resources Forensic Check 25
Resources Forensic Check Comparison 25
Activity Attribute Tab 26
Activity Attribute Forensic Check 26
Activity Attribute Forensic Check Comparison 26
Platform Integration Using Visual Basic for Applications (VBA) 28
General Module 28
Metric Module 28
Phases Module 31
Project Module 32
Ribbon Module 35
RibbonView Module 35
Workbook Module 35
Microsoft Word Custom Report Example 37
Method to Fill the Document with All the Ribbon Views Data 37
Method to Create a Table with All the Metric Information 37
Method to Fill a Table with All the Metric Information 38
Microsoft Excel Custom Report Example 38
Platform Integration Using SAP Crystal Reports 40
Prerequisite 40

Steps to Create a Custom Report 40
Crystal Reports Custom Report Example 44
Using the Acumen API with Other Applications 45

Deltek

Overview

An Application Programming Interface (API) is a set of rules and specifications that software programs can follow to communicate with each other. It serves as an interface between different software programs and facilitates their interaction in much the same way as the user interface facilitates interaction between humans and computers.

The Deltek Acumen API provides a rapid and simple way to communicate with other applications, sharing only the information that is needed and allowing the other applications to manipulate the data in order to construct custom reports, push the data to a web server, insert the data into a database, and so on.

This guide provides information about the Deltek Acumen API and how to integrate with different platforms to create custom reports.

Deltek

Architecture of the Acumen API

The architecture of the Deltek Acumen API is divided into three separate parts or modules:

- **API Configuration file:** This file is used to configure how the API is going to provide information to Acumen.
- **API Data Model:** This specifies the information that is provided through the API and the way in which the data is structured in the exported file.
- **Platform Integration Framework:** This specifies different ways for accessing the information provided by the API.

Workflow

When you click a menu item generated by the Acumen API, the XML generation process begins. After this process is complete, Acumen executes the application that is going to use the XML file to generate a report.

Deltek

API Configuration File

The Acumen API configuration file is based on the XML open standard. You can use it to configure how the API is going to provide information to Acumen. The file specifies:

- The name and description of the Acumen menu item that is generated by the API.
- The view or tab in which the menu item will be located.
- The information that will be accessed when you select the menu item.
- The location of the file where the information will be placed.
- The location of the file that will be run after the export process is complete.

An API configuration file example, **Reportconfig.xml.sample**, is included in the **Acumen » Templates** folder when you install Acumen. You can use this file to view an API configuration file format or edit it to create a custom configuration file.

Add the API Configuration File Location to Acumen

You can store an API configuration file in any local folder; however, Deltek recommends storing it in the **Acumen Fuse » Templates** folder. You need to add the location of the file to Acumen in order for Acumen to access the file.

To add the API configuration file location to Acumen:

1. In Acumen, click  .
2. Click **Deltek Acumen Options**.
3. On the Templates tab of the Deltek Acumen Options dialog box, in the **API Configuration File** field, enter the path to the API configuration file or click  to navigate to the file location.

API Configuration File Requirements

When you create an API configuration file, every report or API call must be placed inside the **ArrayOfReportSetting** tag and must be of the type **ReportSetting**. Every ReportSetting must contain the following tags:

- **Name:** This indicates the menu item title that will display in Acumen.
- **Description:** This indicates the menu item description that will display in Acumen.
- **Template:** This indicates the location of the file that will be executed after the export process is completed.
- **DataFile:** This indicates the location where the file with all the information provided by the API is going to be stored.
- **ExportRibbonViews:** This indicates which ribbon views should have their information exported by the API. The value can be **All**, **Current**, or **None**.
- **ExportLogicAnalyzer:** This indicates which logic analyzers should have their data exported by the API. The value can be **All**, **Current**, or **None**.

- **ExportForensicAnalyzer:** This indicates which forensic reports should have their data exported by the API. The value can be **All**, **Current**, or **None**.
- **ExportMetricLibrary:** This indicates whether the metric library should be exported by the API. The value can be **true** or **false**.
- **ViewLocation:** This indicates the Acumen tab into which the menu item will be placed. The value can be **Projects**, **Analysis**, **Logic**, or **Forensics**.

Editing an API Configuration File

If you add the API configuration file location to Acumen, then:

- Edit the file. You must close and reopen Acumen for the changes to take effect.
- Rename the file. Or if you want to use a different file, you must add the file location to Acumen.

Example of an API Configuration File

This configuration file example produces a **Sample Report** menu item that you can access on the Acumen Projects tab on the Publish menu:

```
<?xml version="1.0"?>
<ArrayOfReportSetting xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ReportSetting>
    <Name>Sample Report</Name>
    <Description>Sample Report Description</Description>
    <Template>C:\SampleDirectory\SampleTemplate.xltm</Template>
    <DataFile>C:\SampleDirectory\SampleDataFile.xml</DataFile>
    <ExportRibbonViews>All</ExportRibbonViews>
    <ExportLogicAnalyzer>Current</ExportLogicAnalyzer>
    <ExportForensicAnalyzer>None</ExportForensicAnalyzer>
    <ExportMetricLibrary>false</ExportMetricLibrary>
    <ViewLocation>Projects</ViewLocation>
  </ReportSetting>
</ArrayOfReportSetting>
```

On the Acumen Projects tab, click **Publish** to see the new menu item in the drop-down menu.

If you change the **<Name>**, **<Description>**, and **<ViewLocation>** lines, the report now displays on the Acumen Analysis tab in the Publish menu and has a new name and description:

```
<?xml version="1.0"?>
<ArrayOfReportSetting xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <ReportSetting>
    <Name>Sample Report Long Name</Name>
```

```
    <Description>Sample Report Long and Elaborate Description For Our New Menu In The
    Analysis Tab</Description>
    <Template>C:\SampleDirectory\SampleTemplate.xltm</Template>
    <DataFile>C:\SampleDirectory\SampleDataFile.xml</DataFile>
    <ExportRibbonViews>All</ExportRibbonViews>
    <ExportLogicAnalyzer>Current</ExportLogicAnalyzer>
    <ExportForensicAnalyzer>None</ExportForensicAnalyzer>
    <ExportMetricLibrary>false</ExportMetricLibrary>
    <ViewLocation>Analysis</ViewLocation>
  </ReportSetting>
</ArrayOfReportSetting>
```

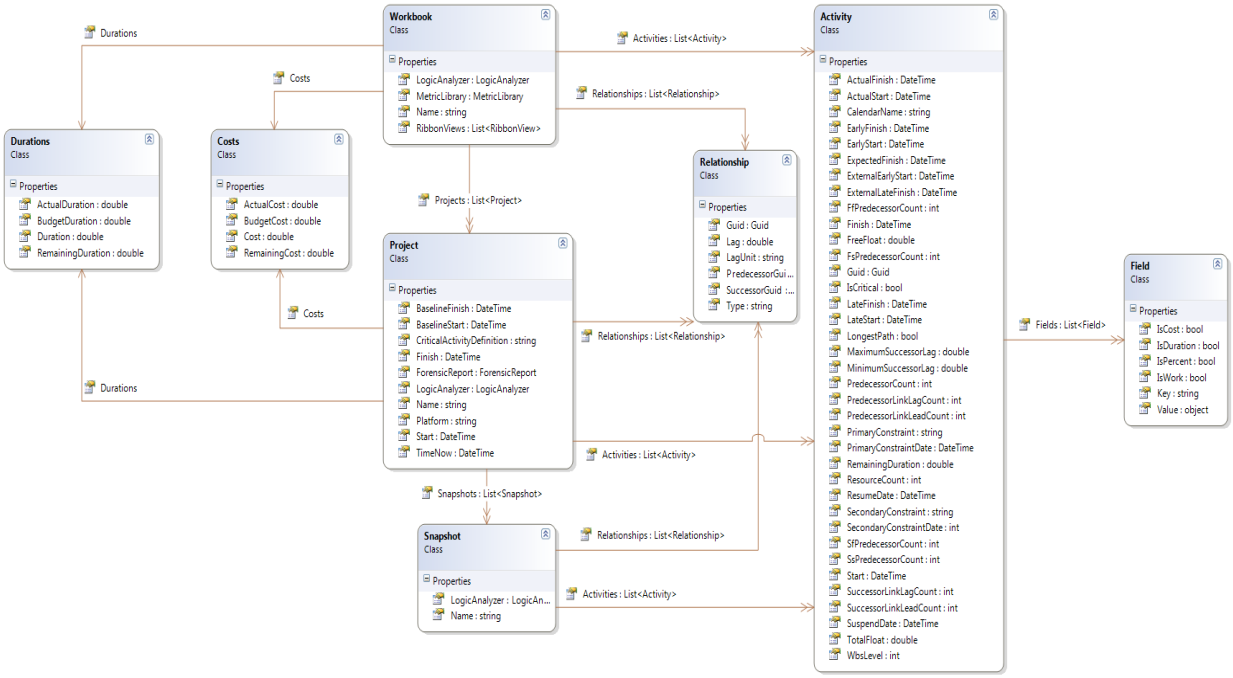
On the Acumen Analysis tab, click **Publish** to see the new menu item in the drop-down menu.

Deltek

API Data Model

The Acumen API Data Model is based on the SML open standard and provides simple and rapid integration with other applications. It is sub-divided into modules which contain multiple classes.

Workbook Module Class Diagram



Workbook

This represents an Acumen workbook.

Field	Type	Description
Name	string	Indicates the workbook name
Costs	Costs	Contains the cost information of the workbook
Durations	Durations	Contains the duration information of the workbook
Activities	List<Activity>	A list of all the activities included inside the workbook
Relationships	List<Relationship>	A list of all the relationships included inside the workbook

Projects	List<Project>	A list of all the projects included inside the workbook
LogicAnalyzer	LogicAnalyzer	Contains all the logic-related information of the workbook
RibbonViews	List<RibbonView>	A list of all the ribbon views included inside the workbook
MetricLibrary	MetricLibrary	Represents the metric library used in the workbook.

Costs

This represents a structure used to store cost information.

Field	Type	Description
ActualCost	double	Indicates the actual cost of the parent
BudgetCost	double	Indicates the budget cost of the parent
Cost	double	Indicates the total cost of the parent
RemainingCost	double	Indicates the remaining cost of the parent

Durations

This represents a structure used to store duration information.

Field	Type	Description
ActualDuration	double	Indicates the actual duration of the parent
BudgetDuration	double	Indicates the budget duration of the parent
Duration	double	Indicates the total duration of the parent
RemainingDuration	double	Indicates the remaining duration of the parent

Deltek

Activity

This represents an Acumen activity.

Field	Type	Description
ActualFinish	DateTime	Indicates the actual finish of the activity
ActualStart	DateTime	Indicates the actual start of the activity
CalendarName	string	Indicates the name of the calendar of the activity
EarlyFinish	DateTime	Indicates the early finish of the activity
EarlyStart	DateTime	Indicates the early start of the activity
ExpectedFinish	DateTime	Indicates the expected finish of the activity
ExternalEarlyStart	DateTime	Indicates the external early start of the activity
ExternalLateFinish	DateTime	Indicates the external late finish of the activity
FfPredecessorCount	int	Indicates the number of Finish-to-Finish predecessors the activity has
Fields	List<Field>	A list of all the non-mandatory fields of the activity
Finish	DateTime	Indicates the finish of the activity
FreeFloat	double	Indicates the free float of the activity
FsPredecessorCount	int	Indicates the number of Finish-to-Start predecessors the activity has
Guid	Guid	Indicates the activity Guid
IsCritical	bool	Indicates if the activity is critical

LateFinish	DateTime	Indicates the late finish of the activity
LateStart	DateTime	Indicates the late start of the activity
LongestPath	bool	Indicates if the activity is inside the longest path
MaximumSuccessorLag	double	Indicates the maximum successor lag for the activity
MinimumSuccessorLag	double	Indicates the minimum successor lag for the activity
PredecessorCount	int	Indicates the number of predecessors the activity has
PredecessorLinkLagCount	int	Indicates the number of predecessors of the activity that have lags
PredecessorLinkLeadsCount	int	Indicates the number of predecessors of the activity that have leads
PrimaryConstraint	string	Indicates the primary constraint of the activity
PrimaryConstraintDate	DateTime	Indicates the primary constraint date for the activity
RemainingDuration	double	Indicates the remaining duration for the activity
ResourceCount	int	Indicates the number of resources assigned to the activity
ResumeDate	DateTime	Indicates the resume date of the activity
SecondaryConstraint	string	Indicates the secondary constraint of the activity
SecondaryConstraintDate	DateTime	Indicates the secondary constraint date for the activity

SfPredecessorCount	int	Indicates the number of Start-to-Finish predecessors the activity has
SsPredecessorCount	int	Indicates the number of Start-to-Start predecessors the activity has
Start	DateTime	Indicates the start date of the activity
SuccessorLinkLagCount	int	Indicates the number of successors of the activity that have lags
SuccessorLinkLeadCount	int	Indicates the number of successors of the activity that have leads
SuspendDate	DateTime	Indicates the suspend date of the activity
TotalFloat	double	Indicates the total float for the activity in minutes
WbsLevel	int	Indicates the WBS level for the activity

Field

This represents an Acumen activity field.

Field	Type	Description
IsCost	bool	Indicates if the field is a cost field
IsDuration	bool	Indicates if the field is a duration field
IsPercent	bool	Indicates if the field is a percentage field
IsWork	bool	Indicates if the field is a work field
Key	string	Indicates the field's key
Value	object	Indicates the field value of the activity

Relationship

This represents an Acumen relationship between two activities.

Field	Type	Description
Guid	Guid	Indicates if the relationship Guid
Lag	double	Indicates the lag for the relationship
LagUnit	string	Indicates the unit in which the lag is represented
PredecessorGuid	Guid	Indicates the predecessor's Guid
SuccessorGuid	Guid	Indicates the successor's Guid
Type	String	Indicates the type of the relationship

Project

This represents an Acumen project.

Field	Type	Description
Activities	List<Activity>	A list of all the activities included inside the project
BaselineFinish	DateTime	Indicates the baseline finish date of the project
BaselineStart	DateTime	Indicates the baseline start date of the project
CriticalActivityDefinition	String	Indicates the criteria for identifying critical activities inside the project
Costs	Costs	Contains the cost information of the project
Durations	Durations	Contains the duration information of the project
Finish	DateTime	Indicates the finish date for the project

ForensicReport	ForensicReport	Contains all the forensic related information of the project
LogicAnalyzer	LogicAnalyzer	Contains all the logic related information of the project
Name	string	Indicates the name of the project
Platform	string	Indicates the source platform of the project
Relationships	List<Relationship>	A list of all the relationships included inside the project
Start	DateTime	Indicates the start date for the project
TimeNow	DateTime	Indicates the status date for the project

Snapshot

This represents an Acumen snapshot.

Field	Type	Description
Activities	List<Activity>	A list of all the activities included inside the snapshot
LogicAnalyzer	LogicAnalyzer	Contains all the logic related information of the snapshot
Name	string	Indicates the name of the snapshot
Relationships	List<Relationship>	A list of all the relationships included inside the snapshot

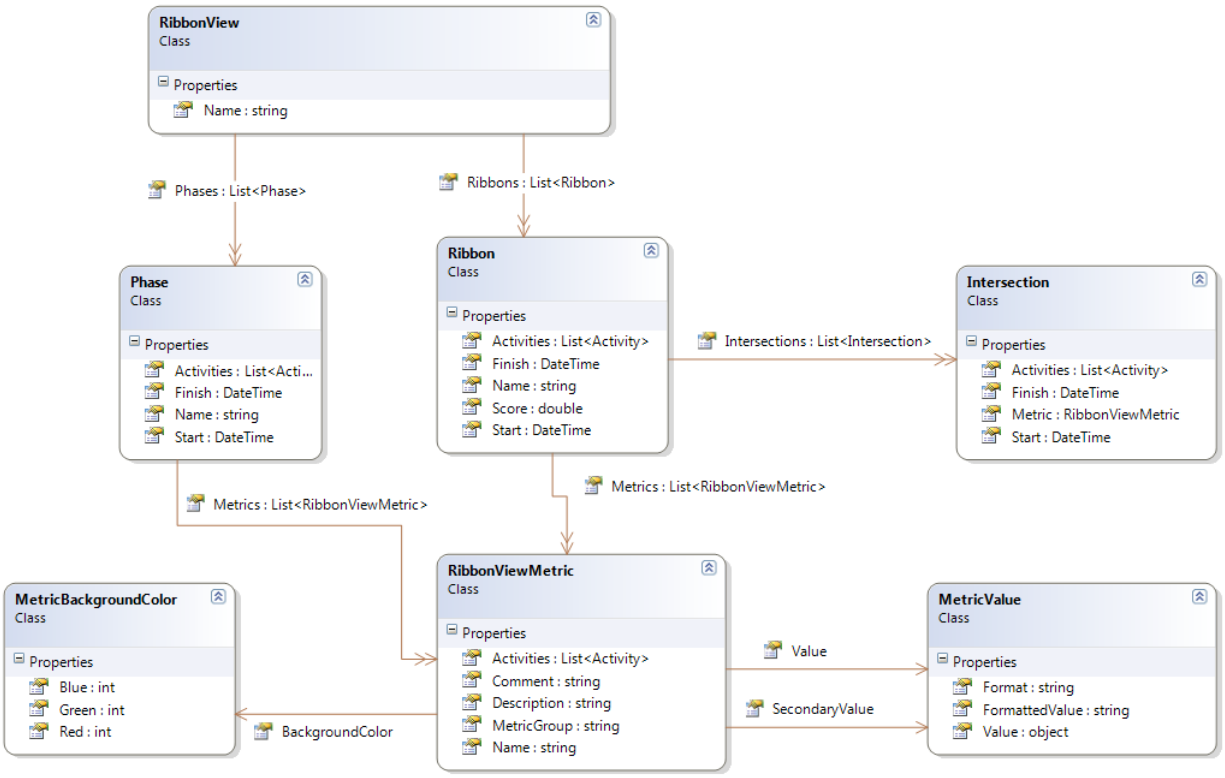
Logic Analyzer

This represents an Acumen Logic Analyzer.

Field	Type	Description
CircularLogic	List<Activity>	A list of all the activities that have circular logic inside the parent
FFRelationships	List<Relationship>	A list of all the FF relationships inside the parent

FSRelationships	List<Relationship>	A list of all the FS relationships inside the parent
Lags	List<Relationship>	A list of all the relationships that have lags inside the parent
Leads	List<Relationship>	A list of all the relationships that have leads inside the parent
LogicOnSummaries	List<Relationship>	A list of all the relationships associated to summary activities
OpenEnds	List<Activity>	A list of all the activities with open ends
OpenFinish	List<Activity>	A list of all the activities with an open finish
OpenStart	List<Activity>	A list of all the activities with an open start
OutOfSequenceLogic	List<Relationship>	A list of all the relationships that are out of sequence
RedundantLogic	List<Relationship>	A list of all the relationships that are redundant
ReverseLogic	List<Relationship>	A list of all the reverse logic relationships
SFRelationships	List<Relationship>	A list of all the SF relationships inside the parent
SSRelationships	List<Relationship>	A list of all the SS relationships inside the parent

Ribbon View Module Class Diagram



Ribbon View

This represents an Acumen ribbon view.

Field	Type	Description
Name	string	Indicates the name of the ribbon view
Phases	List<Phase>	A list of all the phases inside the ribbon view
Ribbons	List<Ribbon>	A list of all the ribbons inside the ribbon view

Phase

This represents an Acumen ribbon view phase.

Field	Type	Description

Activities	List<Activity>	A list of all the activities inside the phase
Finish	DateTime	Indicates the finish date of the phase
Name	string	Indicates the name of the phase
Start	DateTime	Indicates the start date of the phase
Metrics	List<RibbonViewMetric>	A list of all the metric cells inside the phase

Ribbon

This represents an Acumen ribbon view ribbon.

Field	Type	Description
Activities	List<Activity>	A list of all the activities inside the ribbon
Finish	DateTime	Indicates the finish date of the ribbon
Intersections	List<Intersection>	A list of all the intersections inside the ribbon
Name	string	Indicates the name of the ribbon
Score	double	Indicates the score of the ribbon
Start	DateTime	Indicates the start date of the ribbon
Metrics	List<RibbonViewMetric>	A list of all the metric cells inside the ribbon

Intersection

This represents an Acumen ribbon view intersection.

Field	Type	Description

Activities	List<Activity>	A list of all the activities inside the phase
Finish	DateTime	Indicates the finish date of the phase
Start	DateTime	Indicates the start date of the phase
Metric	RibbonViewMetric	Contains all the metric information for the intersection cell

Ribbon View Metric

This represents an Acumen ribbon view metric.

Field	Type	Description
Activities	List<Activity>	A list of all the activities inside the ribbon view metric
BackgroundColor	MetricBackgroundColor	Contains the RGB components for the background color of the metric cell
Comment	string	Indicates the comment for the ribbon view metric
Description	string	Indicates the description for the ribbon view metric
MetricGroup	string	Indicates which metric group the ribbon view metric belongs to
Name	string	Indicates the name of the ribbon view metric
Value	MetricValue	Contains all the primary value information for the metric
SecondaryValue	MetricValue	Contains all the secondary value information for the metric

Metric Value

This represents an Acumen ribbon view metric value.

Field	Type	Description
Format	string	Indicates the formula format for the value
FormattedValue	string	Indicates the primary value after applying the formula format
Value	object	Indicates the value for the metric

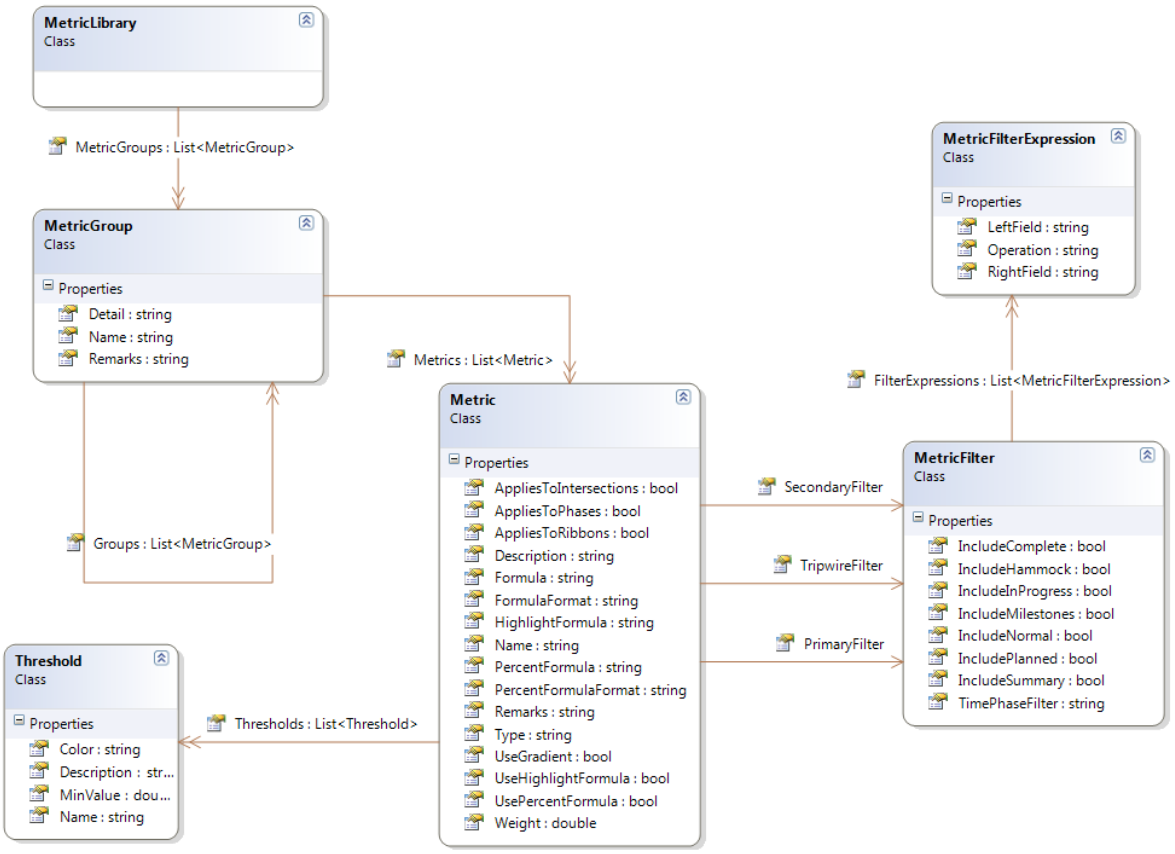
Metric Background Color

This represents an Acumen ribbon view metric background metric.

Field	Type	Description
Blue	int	Indicates the blue component for the background color
Green	int	Indicates the green component for the background color
Red	int	Indicates the red component for the background color

Deltek

Metric Library Module Class Diagram



Metric Library

This represents an Acumen metric group.

Field	Type	Description
MetricGroups	List<MetricGroup>	A list of all the metric groups inside the metric library

Metric Group

This represents an Acumen metric group.

Field	Type	Description
Detail	string	Indicates the details for the metric group

Groups	List<MetricGroup>	A list of all the metric sub-groups inside the metric group
Metrics	List<Metric>	A list of all the metrics inside the metric group
Name	string	Indicates the name of the metric group
Remarks	string	Indicates the remarks for the metric group

Metric

This represents an Acumen metric.

Field	Type	Description
AppliesToIntersections	bool	Indicates if the metric applies to intersections
AppliesToPhases	bool	Indicates if the metric applies to phases
AppliesToRibbons	bool	Indicates if the metric applies to ribbons
Description	string	Indicates the description for the metric
Formula	string	Indicates the primary formula for the metric
FormulaFormat	string	Indicates the primary formula format for the metric
HighlightFormula	string	Indicates the tripwire formula for the metric
Name	string	Indicates the name of the metric
PercentFormula	string	Indicates the percent formula for the metric
PercentFormulaFormat	string	Indicates the percent formula format for the metric

Remarks	string	Indicates the remarks for the metric
Type	string	Indicates the type of metric
UseGradient	bool	Indicates if the metric uses a gradient for the background colors
UseHighlightFormula	bool	Indicates if the highlight formula is in use for the metric
UsePercentFormula	bool	Indicates if the percent formula is in use for the metric
Weight	double	Indicates the weight of the metric for the metric group

Threshold

This represents an Acumen metric threshold.

Field	Type	Description
Color	string	Indicates the color if the metric value falls inside this threshold
Description	string	Indicates a description for the threshold
Metrics	List<Metric>	A list of all the metrics inside the metric group
Name	string	Indicates the name of the metric group
Remarks	string	Indicates the remarks for the metric group

Metric Filter

This represents an Acumen metric filter.

Field	Type	Description
IncludeComplete	bool	Indicates if the metric should filter the completed activities

IncludeHammock	bool	Indicates if the metric should filter the hammock activities
IncludeInProgress	bool	Indicates if the metric should filter the activities that are in progress
IncludeMilestones	bool	Indicates if the metric should filter the milestone activities
IncludeNormal	bool	Indicates if the metric should filter the normal activities
IncludePlanned	bool	Indicates if the metric should filter the planned activities
IncludeSummary	bool	Indicates if the metric should filter the summary activities
TimePhaseFilter	string	Indicates the time phase filter for the metric

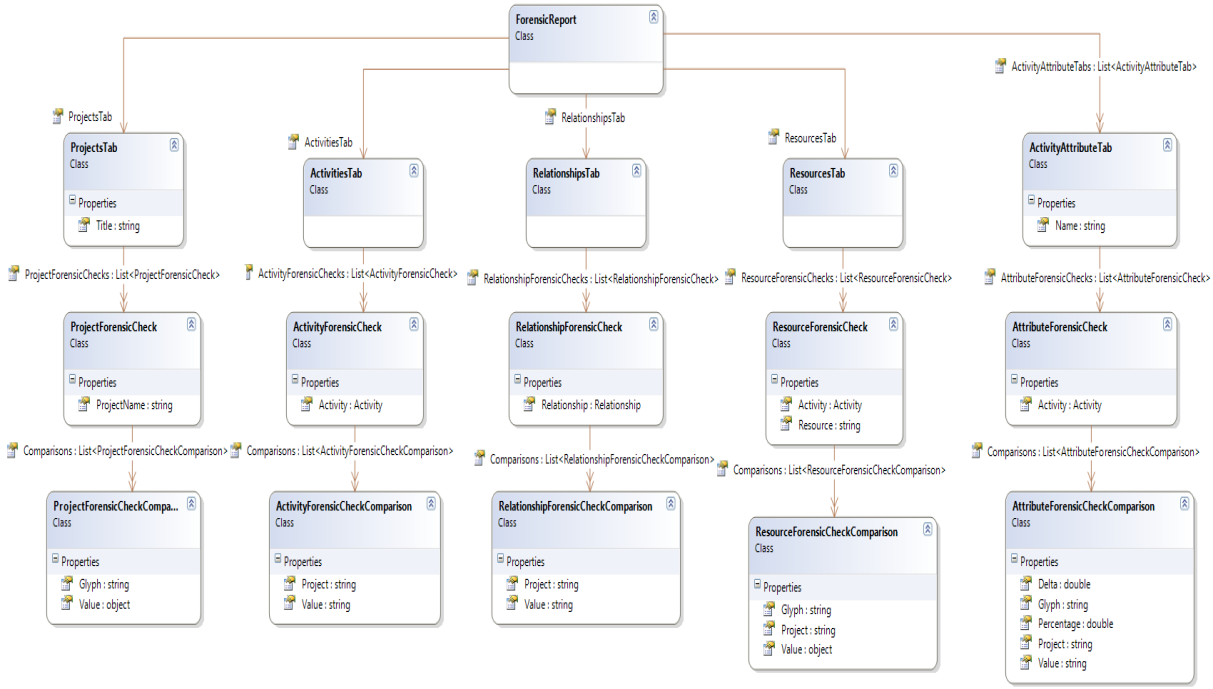
Metric Filter Expression

This represents an Acumen metric filter expression.

Field	Type	Description
LeftField	string	Indicates the left field in the expression
RightField	string	Indicates the right field in the expression
Operation	string	Indicates the operation that should be applied between the two fields

Deltek

Forensic Report Module Class Diagram



Forensic Report

This represents an Acumen Forensic Report. This contains comparison information between one project and its snapshots.

Field	Type	Description
ProjectsTab	ProjectsTab	Contains all the information for the projects tab inside the forensic report
ActivitiesTab	ActivitiesTab	Contains all the information for the activities tab inside the forensic report
RelationshipsTab	RelationshipsTab	Contains all the information for the relationships tab inside the forensic report
ResourcesTab	ResourcesTab	Contains all the information for the resources tab inside the forensic report
ActivititAttributeTabs	List<ActivityAttributeTab>	A list of all the activity attribute tabs inside the forensic report

Projects Tab

This represents a projects tab inside the Acumen Forensic Report.

Field	Type	Description
Title	string	Indicates the title of the tab
ProjectForensicChecks	List<ProjectForensicCheck>	A list of all the project forensic checks contained inside the projects tab

Project Forensic Check

This represents a project forensic check inside a Projects tab.

Field	Type	Description
ProjectName	string	Indicates the name of the project
Comparisons	List<ProjectForensicCheckComparison>	A list of all the comparisons inside the project forensic check

Project Forensic Check Comparison

This represents a project forensic check comparison inside each project forensic check.

Field	Type	Description
Glyph	string	Indicates how the value changed compared to the main project
Value	Object	Indicates the value for the current project

Activities Tab

This represents an activities tab inside the Acumen Forensic Report.

Field	Type	Description
ActivityForensicChecks	List<ActivityForensicCheck>	A list of all the activity forensic checks contained inside the activities tab

Activity Forensic Check

This represents an activity forensic check inside an activities tab.

Field	Type	Description
Activity	Activity	Indicates the activity that is currently being compared
Comparisons	List<ActivityForensicCheckComparison>	A list of all the comparisons inside the activity forensic check

Activity Forensic Check Comparison

This represents an activity forensic check comparison inside each activity forensic check.

Field	Type	Description
Project	string	Indicates the project that contains the activity currently being compared
Value	string	Indicates the value for the current project

Relationships Tab

This represents a relationships tab inside the Acumen Forensic Report.

Field	Type	Description
RelationshipForensicChecks	List<RelationshipForensicCheck>	A list of all the relationship forensic checks contained inside the relationships tab

Relationships Forensic Check

This represents a relationship forensic check inside a relationships tab.

Field	Type	Description
Relationship	Relationship	Indicates the relationship that is currently being compared
Comparisons	List<RelationshipForensicCheckComparison>	A list of all the comparisons inside the relationship forensic check

Relationship Forensic Check Comparison

This represents a relationship forensic check comparison inside each relationship forensic check.

Field	Type	Description
Project	String	Indicates the project that contains the relationship currently being compared
Value	string	Indicates the value for the current project

Resources Tab

This represents a resources tab inside the Acumen Forensic Report.

Field	Type	Description
ResourceForensicChecks	List<ResourceForensicCheck>	A list of all the resource forensic checks contained inside the resources tab

Resources Forensic Check

This represents a resource forensic check inside a resources tab.

Field	Type	Description
Activity	Activity	Indicates the activity that contains the resource that is currently being compared
Resource	String	Indicates the resource that is currently being compared
Comparisons	List<ResourceForensicCheckComparison>	A list of all the comparisons inside the resource forensic check

Resources Forensic Check Comparison

This represents a resource forensic check comparison inside each resource forensic check.

Field	Type	Description
Project	string	Indicates the project that contains the resource currently being compared

Glyph	string	Indicates how the value changed compared to the main project
Value	string	Indicates the value for the current project

Activity Attribute Tab

This represents an activity attribute tab inside the Acumen Forensic Report.

Field	Type	Description
Name	string	Indicates the title of the tab
AttributeForensicChecks	List<AttributeForensicCheck>	A list of all the activity attribute forensic checks contained inside the activity attribute tab

Activity Attribute Forensic Check

This represents an activity attribute forensic check inside an activity attribute tab.

Field	Type	Description
Activity	Activity	Indicates the activity that is currently being compared
Comparisons	List<AttributeForensicCheckComparison>	A list of all the comparisons inside the activity attributes forensic check

Activity Attribute Forensic Check Comparison

This represents an activity attribute forensic check comparison inside each activity attribute forensic check.

Field	Type	Description
Project	string	Indicates the project that contains the resource currently being compared
Glyph	string	Indicates how the value changed compared to the main project

Value	string	Indicates the value for the current project
Delta	double	Indicates the difference of the main project value compared to the current one
Percentage	Double	Indicates the percentage that the delta represents compared to the main value

Deltek

Platform Integration Using Visual Basic for Applications (VBA)

This topic describes how to integrate with Visual Basic for Applications (VBA) to create custom reports.

> **Note:** Acumen installs the module (.bas) files in the **%InstallationDirectory%\API Modules** folder. Knowledge of Visual Basic for Applications (VBA) is a prerequisite for creating custom reports.

General Module

DesiredNode Method

The DesiredNode method obtains a node with the specified name.

```
Function DesiredNode (xParentNode As MSXML2.IXMLDOMNode, nodeName As String) As
MSXML2.IXMLDOMNode
```

Parameters

`xParentNode`

`Type: MSXML2.IXMLDOMNode`

This is the node in which the method will look for a child node.

`nodeName`

`Type: System.String`

This is the string that specifies the name of the child node to look for.

Return Value

`Type: MSXML2.IXMLDOMNode`

This is an XML Node child to the parent node with the specified name.

Metric Module

MetricName Method

The MetricName method obtains the name of the metric.

```
Function MetricName (metricNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`metricNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a metric.

Return Value

`Type: System.String`

This is a string that indicates the name of the metric.

MetricValue Method

The MetricValue method obtains the value of the metric.

```
Function MetricValue (metricNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`metricNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a metric.

Return Value

`Type: System.String`

This is a string that indicates the primary value of the metric.

MetricSecondaryValue Method

The MetricSecondaryValue method obtains the secondary value of the metric.

```
Function MetricSecondaryValue (metricNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`metricNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a metric.

Return Value

`Type: System.String`

This is a string that indicates the secondary value of the metric.

MetricPrimaryFormulaFormat Method

The MetricPrimaryFormulaFormat method obtains the primary formula format of the metric.

```
Function MetricPrimaryFormulaFormat (metricNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`metricNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a metric.

Return Value

`Type: System.String`

This is a string that indicates the primary formula format of the metric.

MetricSecondaryFormulaFormat Method

The MetricSecondaryFormulaFormat method obtains the secondary formula format of the metric.

```
Function MetricSecondaryFormulaFormat (metricNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`metricNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a metric.

Return Value

`Type: System.String`

This is a string that indicates the secondary formula format of the metric.

MetricBackground Method

The MetricBackground method obtains the background color of the metric.

```
Function MetricBackground (metricNode As MSXML2.IXMLDOMNode) As Long
```

Parameters

`metricNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a metric.

Return Value

`Type: System.Long`

This is a long value that represents the RGB number for the background color.

MetricComment Method

The MetricComment method obtains the comment of the metric.

```
Function MetricComment (metricNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`metricNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a metric.

Return Value

`Type: System.String`

This is a string that indicates the comment of the metric.

MetricRemarks Method

The MetricRemarks method obtains the remarks of the metric.

```
Function MetricRemarks (metricNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`metricNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a metric.

Return Value

`Type: System.String`

This is a string that indicates metric remarks.

Phases Module

PhaseStart Method

The PhaseStart method obtains the phase start date.

Function PhaseStart (phaseNode ***As*** MSXML2.IXMLDOMNode) ***As*** String

Parameters

`phaseNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a phase.

Return Value

`Type: System.String`

This is a string that indicates the phase start date.

PhaseFinish Method

The PhaseFinish method obtains the phase finish date.

Function PhaseFinish (phaseNode ***As*** MSXML2.IXMLDOMNode) ***As*** String

Parameters

`phaseNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a phase.

Return Value

`Type: System.String`

This is a string that indicates the phase finish date.

PhaseName Method

The PhaseName method obtains the name of the phase.

Function PhaseName (phaseNode *As* MSXML2.IXMLDOMNode) *As* String

Parameters

`phaseNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a phase.

Return Value

`Type: System.String`

This is a string that indicates the phase name.

Project Module

ProjectName Method

The ProjectName method obtains the name of the project.

Function `ProjectName (projectNode` *As* `MSXML2.IXMLDOMNode)` *As* `String`

Parameters

`projectNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a project.

Return Value

`Type: System.String`

This is a string that indicates the project name.

ProjectStart Method

The ProjectStart method obtains the project start date.

Function `ProjectStart (projectNode` *As* `MSXML2.IXMLDOMNode)` *As* `String`

Parameters

`projectNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a project.

Return Value

`Type: System.String`

This is a string that indicates the project start date.

ProjectFinish Method

The ProjectFinish method obtains the project finish date.

```
Function ProjectFinish (projectNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`projectNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a project.

Return Value

`Type: System.String`

This is a string that indicates the project finish date.

ProjectTimeNow Method

The ProjectTimeNow method obtains the project time now date.

```
Function ProjectTimeNow (projectNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`projectNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a project.

Return Value

`Type: System.String`

This is a string that indicates the project time now date.

ProjectActualCost Method

The ProjectActualCost method obtains the project actual cost.

```
Function ProjectActualCost (projectNode As MSXML2.IXMLDOMNode) As Double
```

Parameters

`projectNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a project.

Return Value

`Type: System.Double`

This is a double that indicates the project actual cost.

ProjectRemainingCost Method

The ProjectRemainingCost method obtains the project remaining cost.

```
Function ProjectRemainingCost (projectNode As MSXML2.IXMLDOMNode) As Double
```

Parameters

`projectNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a project.

Return Value

`Type: System.Double`

This is a double that indicates the project remaining cost.

ProjectBudgetCost Method

The ProjectBudgetCost method obtains the project budget cost.

```
Function ProjectBudgetCost (projectNode As MSXML2.IXMLDOMNode) As Double
```

Parameters

`projectNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a project.

Return Value

`Type: System.Double`

This is a double that indicates the project budget cost.

ProjectBaselineStart Method

The ProjectBaselineStart method obtains the project baseline start date.

```
Function ProjectBaselineStart (projectNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`projectNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a project.

Return Value

`Type: System.String`

This is a string that represents the project baseline start date.

ProjectBaselineFinish Method

The ProjectBaselineFinish method obtains the project baseline finish date.

```
Function ProjectBaselineFinish (projectNode As MSXML2.IXMLDOMNode) As String
```

Parameters

`projectNode`

Type: MSXML2.IXMLDOMNode

This is an XML node that corresponds to a project.

Return Value

Type: System.String

This is a string that represents the project baseline finish date.

Ribbon Module

GetRibbonName Method

The GetRibbonName method obtains the ribbon name.

```
Function GetRibbonName (xRibbonNode As MSXML2.IXMLDOMNode) As String
```

Parameters

xRibbonNode

Type: MSXML2.IXMLDOMNode

This is an XML node that corresponds to a ribbon.

Return Value

Type: System.String

This is a string that represents the name of the ribbon.

RibbonView Module

GetRibbonViewName Method

The GetRibbonViewName method obtains the ribbon view name.

```
Function GetRibbonViewName (xRibbonViewNode As MSXML2.IXMLDOMNode) As String
```

Parameters

xRibbonViewNode

Type: MSXML2.IXMLDOMNode

This is an XML node that corresponds to a ribbon view.

Return Value

Type: System.String

This is a string that represents the name of the ribbon view.

Workbook Module

WorkbookTotalCost Method

The WorkbookTotalCost method obtains the workbook total cost.

```
Function WorkbookTotalCost (xWorkbookNode As MSXML2.IXMLDOMNode) As Double
```

Parameters

`xWorkbookNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a workbook.

Return Value

`Type: System.Double`

This is a double that indicates the workbook total cost.

WorkbookActualCost Method

The WorkbookActualCost method obtains the workbook actual cost.

```
Function WorkbookActualCost (xWorkbookNode As MSXML2.IXMLDOMNode) As Double
```

Parameters

`xWorkbookNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a workbook.

Return Value

`Type: System.Double`

This is a double that indicates the workbook actual cost.

WorkbookBudgetCost Method

The WorkbookBudgetCost method obtains the workbook budget cost.

```
Function WorkbookBudgetCost (xWorkbookNode As MSXML2.IXMLDOMNode) As Double
```

Parameters

`xWorkbookNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a workbook.

Return Value

`Type: System.Double`

This is a double that indicates the workbook budget cost.

WorkbookRemainingCost Method

The WorkbookRemainingCost method obtains the workbook remaining cost.

```
Function WorkbookRemainingCost (xWorkbookNode As MSXML2.IXMLDOMNode) As Double
```

Parameters

`xWorkbookNode`

`Type: MSXML2.IXMLDOMNode`

This is an XML node that corresponds to a workbook.

Return Value

`Type: System.Double`

This is a double that indicates the workbook remaining cost.

Microsoft Word Custom Report Example

This example creates a Microsoft Word custom report that automatically creates a table that displays the list of metrics and their values (primary and secondary) for all the ribbon views.

> **Note:** Knowledge of Visual Basic for Applications (VBA) is a prerequisite for using this example and for creating custom reports.

Method to Fill the Document with All the Ribbon Views Data

```
Sub FillDocumentWithRibbonViewsData(ribbonViewsNode As MSXML2.IXMLDOMNode)
    Dim ribbonViewNode As MSXML2.IXMLDOMNode
  'Loop through all the different ribbon views and call the AddMetricTable method.
    For Each ribbonViewNode In ribbonViewsNode.ChildNodes
        Dim ribbonNode As MSXML2.IXMLDOMNode
        Set ribbonNode = DesiredNode(ribbonViewNode, "Ribbons").ChildNodes(0)
        Call AddMetricTable(ribbonNode)
    Next
End Sub
```

Method to Create a Table with All the Metric Information

```
Sub AddMetricTable(ribbonNode As MSXML2.IXMLDOMNode)
    If MetricQuantity(ribbonNode) <> 0 Then
        Dim metricTable As table
    'Create table, rows = number of metrics, 3 columns
        Set metricTable = ActiveDocument.Tables.Add(Range:=Selection.Range,
NumRows:=MetricQuantity(ribbonNode), NumColumns:=3)
    'Fill the table just created
        Call FillMetricTable(DesiredNode(ribbonNode, "Metrics"), metricTable)
        metricTable.Select
```

```
    'Move the selection to the end of the table
        Selection.Collapse WdCollapseDirection.wdCollapseEnd
        Selection.TypeParagraph
    End If
End Sub
```

Method to Fill a Table with All the Metric Information

```
Sub FillMetricTable(metricsNode As MSXML2.IXMLDOMNode, metricTable As table)
    Dim i As Integer
  'Loop through all the different metrics.
    For i = 0 To metricsNode.ChildNodes.Length - 1
    'Add the metric name
        metricTable.Cell(i + 1, 0).Range.Text = metricName(metricsNode.ChildNodes(i))
    'Add the primary value
        metricTable.Cell(i + 1, 1).Range.Text =
MetricFormattedValue(metricsNode.ChildNodes(i))
    'Add the secondary value
        metricTable.Cell(i + 1, 2).Range.Text =
MetricFormattedSecondaryValue(metricsNode.ChildNodes(i))
    Next i
End Sub
```

These functions produce the following result:

Open Ends	0	20%
Logic Density™	1.9	
Critical	0	54%
Soft Constraints	0	5%
Hard Constraints	0	7%
High Float	0	32%
Negative Float	0	15%
High Duration	0	13%
Number of Lags	0	54%
Max Lag		

Microsoft Excel Custom Report Example

This example creates a Microsoft Excel custom report that automatically creates a table that displays start and finish for each of the phases in the current ribbon view.

Note: Knowledge of Visual Basic for Applications (VBA) is a prerequisite for using this example and for creating custom reports.

```
Sub FillSheet (costRibbonViewNode As MSXML2.IXMLDOMNode)
    Dim offset As range
    Set offset = Sheet1.range("A2:A2")
    Dim phasesNode As MSXML2.IXMLDOMNode
    Set phasesNode = DesiredNode(costRibbonViewNode, "Phases")
    Dim i As Integer
    Dim budgetCumulative As Double
    budgetCumulative = 0
    Dim forecastCumulative As Double
    forecastCumulative = 0
    Dim actualCumulative As Double
    actualCumulative = 0
    'Loop through all the different phases
    For i = 0 To phasesNode.ChildNodes.Length - 1
        Dim phaseNode As MSXML2.IXMLDOMNode
        Set phaseNode = phasesNode.ChildNodes(i)
    'Add the phase name
        offset.offset(i, 0).Value = PhaseName(phaseNode)
    'Add the phase start
        offset.offset(i, 1).Value = PhaseStart(phaseNode)
    'Add the phase finish
        offset.offset(i, 2).Value = PhaseFinish(phaseNode)
    Next i
End Sub
```

Deltek

Platform Integration Using SAP Crystal Reports

You do not need any programming knowledge to create custom reports using Crystal Reports and the Acumen API.

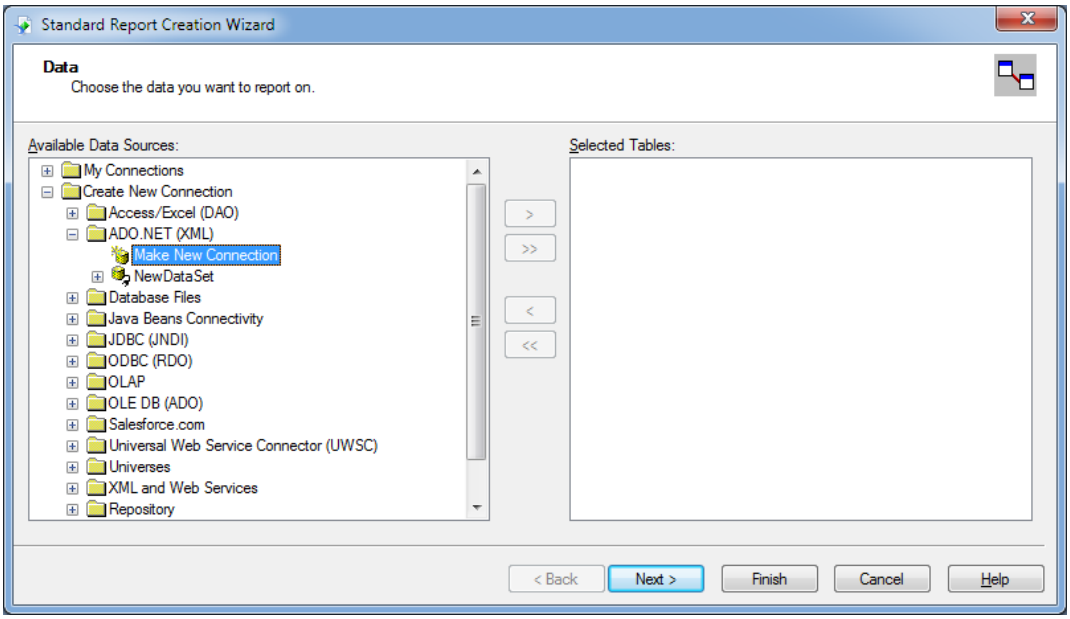
Prerequisite

In order to create a report, you need to have an XML file created by the Acumen API.

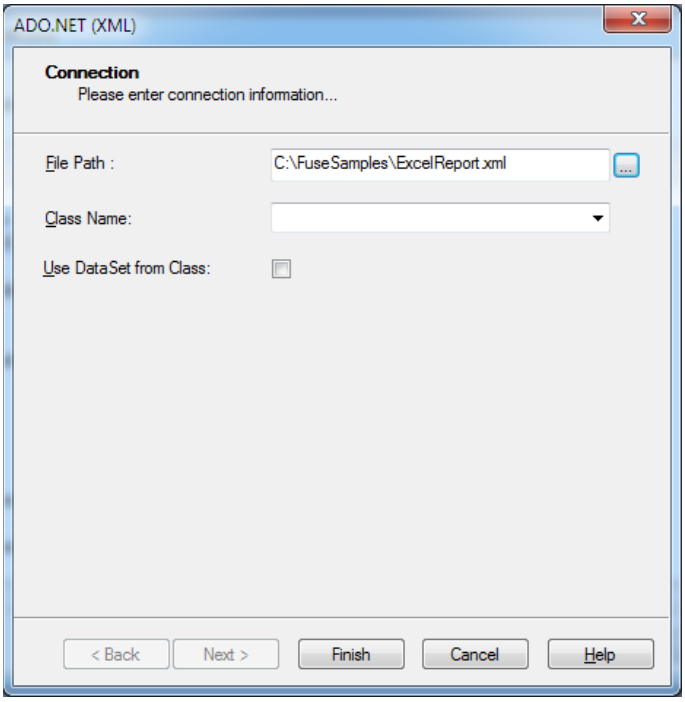
> **Attention:** See API Configuration File for more information.

Steps to Create a Custom Report

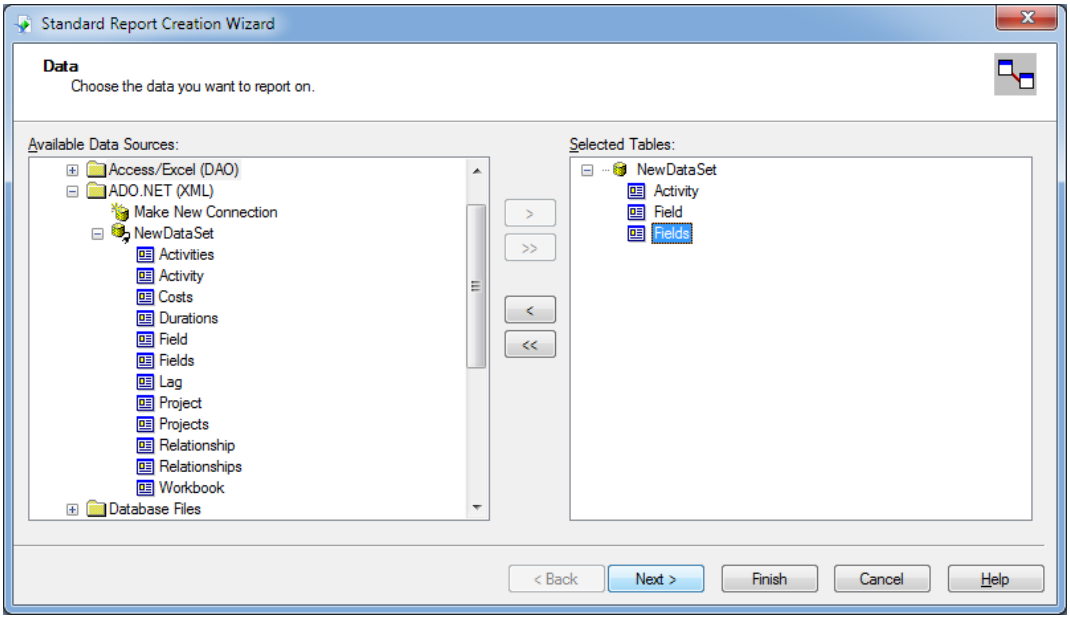
1. Open Crystal Reports and create a new report to start the Standard Report Creation wizard.
2. On the Data screen, select **Create New Connection » ADO.NET (XML) » Make New Connection**.



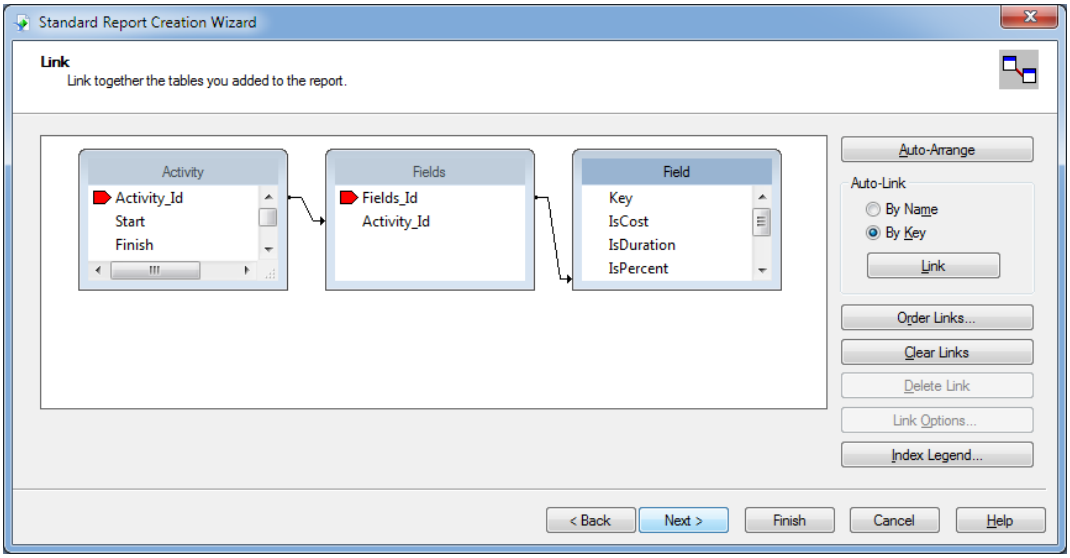
3. On the ADD.NET (XML) dialog box, in the **File Path** field, navigate to or enter the path to the XML file created by the Acumen API.



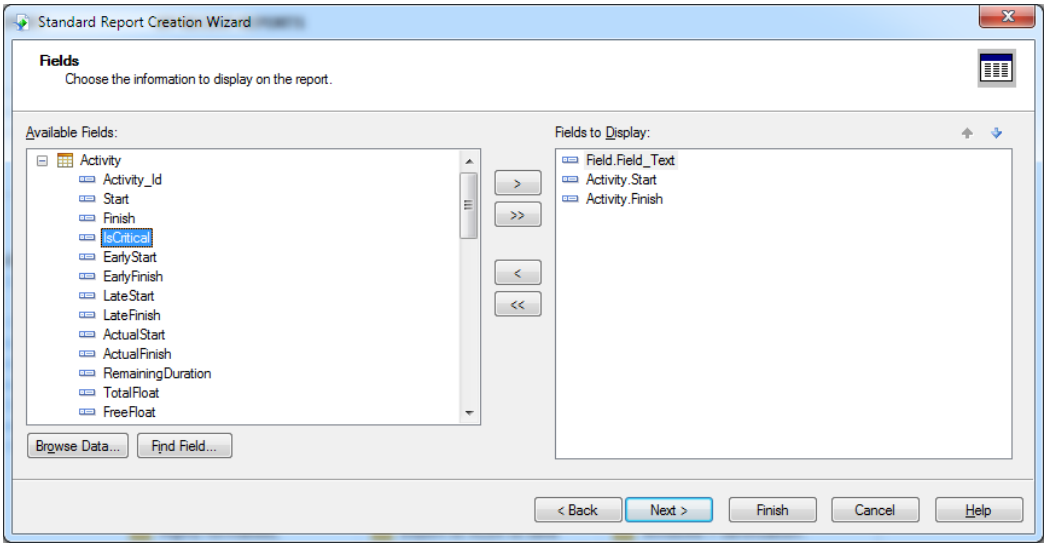
4. Click **Finish**.
5. On the Data screen, add the tables needed for the report to the Selected Tables pane and click **Next**.



If the tables are related, Crystal Reports automatically constructs the relationships and displays them on the Link screen.

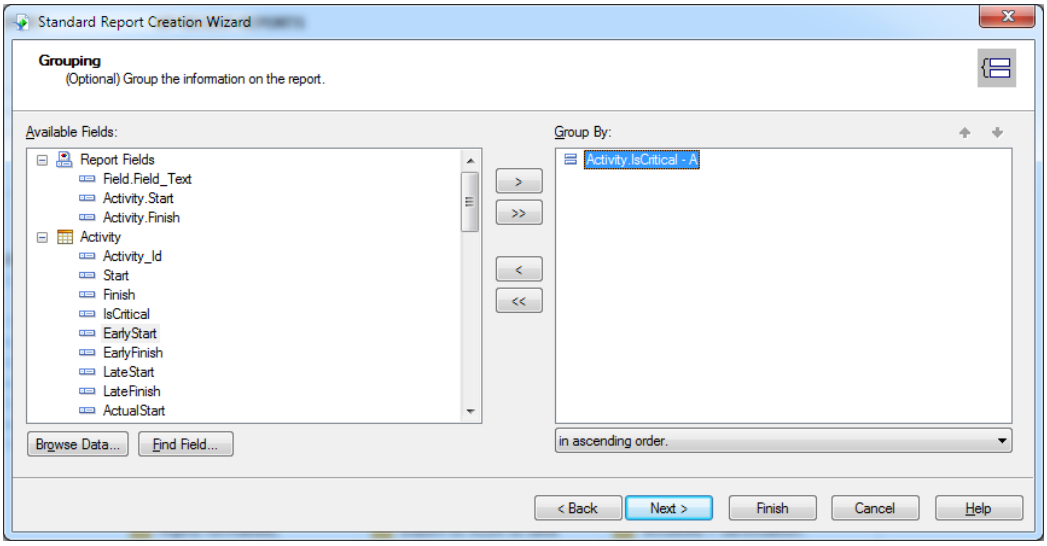


6. Click **Next**.
7. On the Fields screen, add the fields you want to display on the report to the Fields to Display pane and click **Next**.



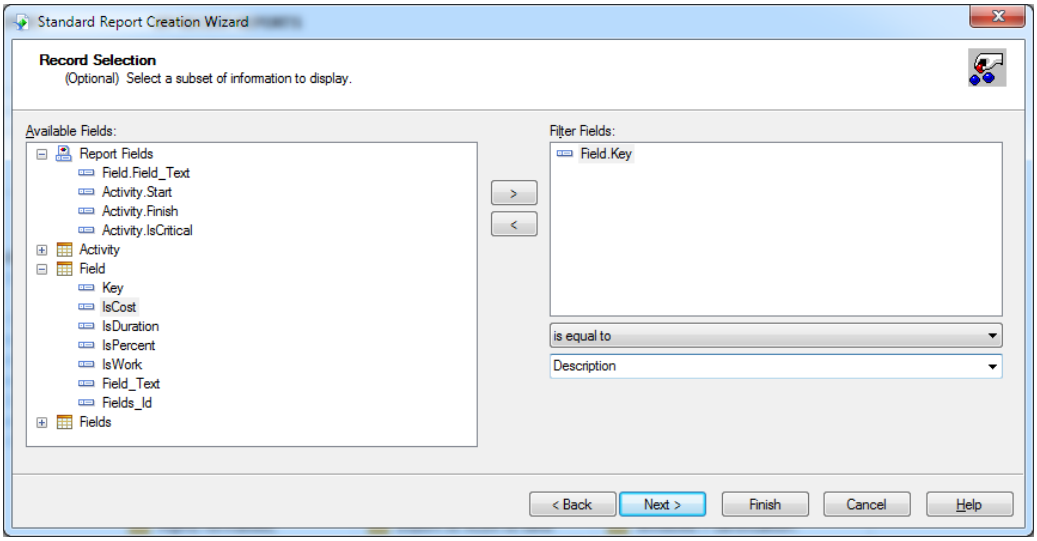
8. On the Grouping screen, add any grouping options you want on the report to the Group By pane.

   For example, you can group the activities depending on their **IsCritical** value.



9. Use the drop-down field below the Group By pane to select the order in which you want the fields listed.
10. Click **Next**.
11. On the Summaries screen, click **Next**.
12. On the Record Selection screen, do one of the following:
    - If you do not want to add any filters, click **Next**.
    - To add a filter to the report, add a field to the Filter Fields pane and select conditions from the drop-down lists, and then click **Next** and **Finish**.

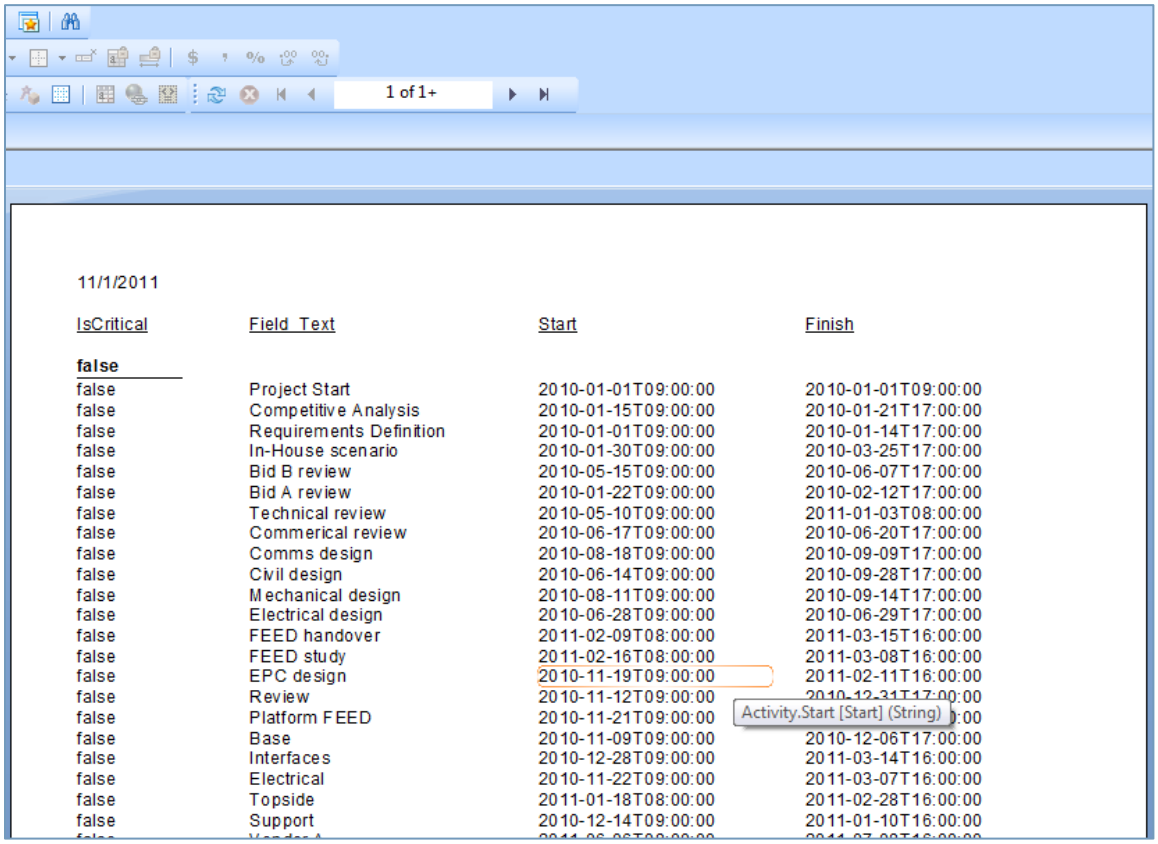
      For example, to create a filter so that the only activity field shown on the report is the description, add **Field Key** to the Filter Fields pane and select **is equal to** and **Description** conditions.



Deltek

Crystal Reports Custom Report Example

The screenshot below is an example of a Crystal Reports custom report.



Deltek

Using the Acumen API with Other Applications

You can use the Acumen API with other applications to upload information to the web, create custom dashboards, push information to Microsoft SharePoint, push data into a database, and so on. Any application that has the ability to read an XML file can use the Acumen API.