



# Tutorial : Event Driven Workflow

## Contents

About this tutorial.....	3
Getting Started.....	4
Designing the Check Baseline Workflow.....	8
Associating the Event Driven Workflow with a Microsoft Project Server event.....	21
Next Steps.....	22

## About this tutorial

Event driven workflows provide the ability to associate Nintex workflows with Microsoft Project Server events. Nintex workflows can be started based on events that happen within Project Server.


This tutorial will describe how to retrieve the enterprise resources that have a Booking Type of Proposed, which have been added as a resource to the project plan every time it is published.

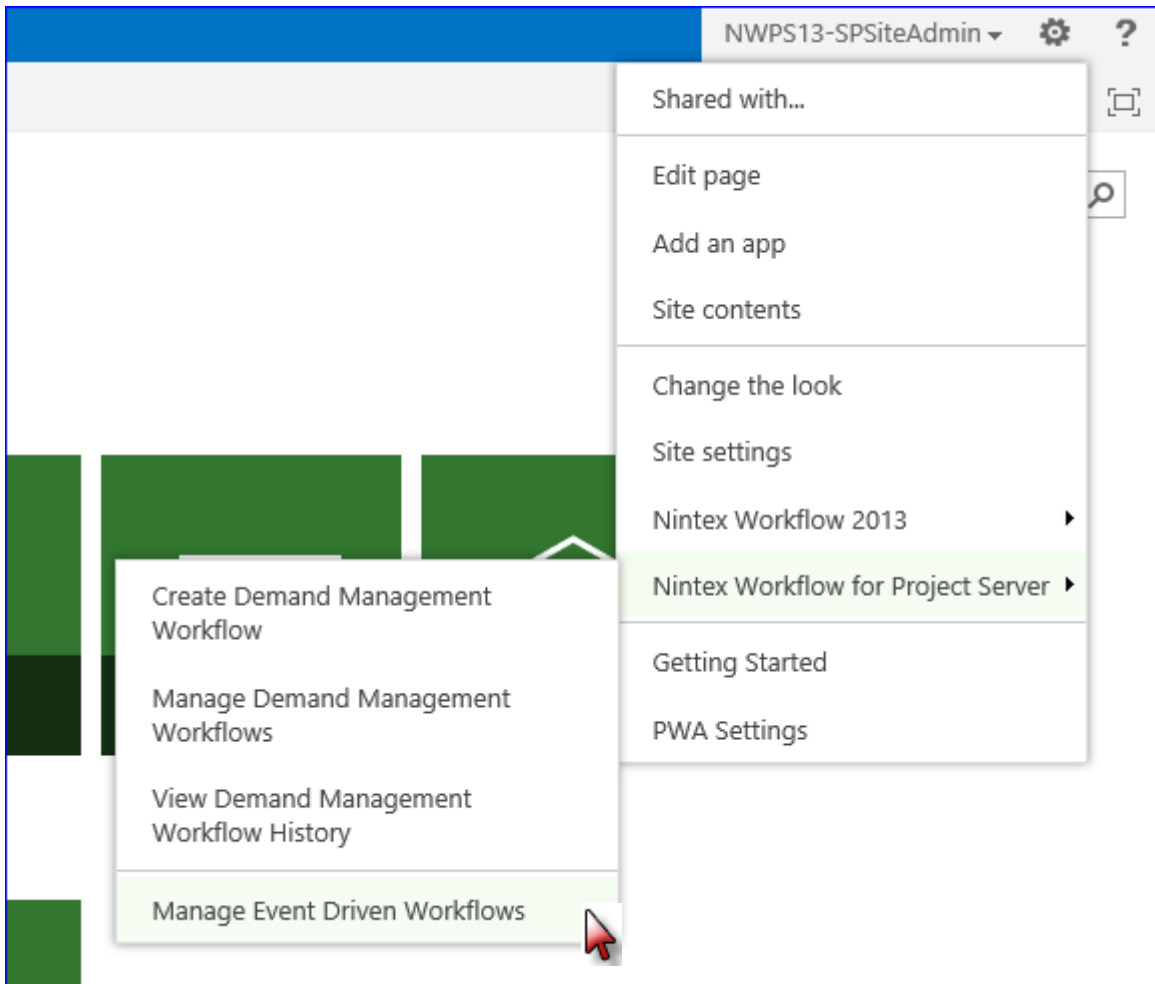
After completion of this tutorial, you will know how to create an event driven workflow, use the *Retrieve event data* action within the workflow, and associate the workflow to a Microsoft Project Server event.

This tutorial assumes the user has basic knowledge of designing workflows in Project Server using Nintex Workflow for Project Server 2013. If not, please refer to tutorials available at [connect.nintex.com](http://connect.nintex.com) for more information.

## Getting Started

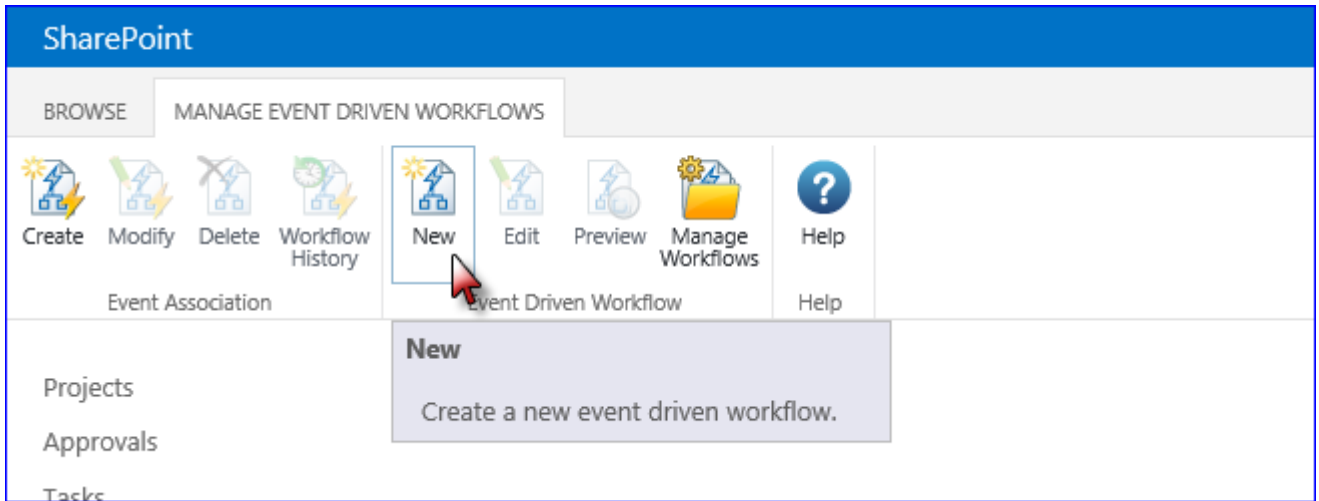
Open Nintex Workflow for Project Server 2013 workflow designer and create a new event driven workflow.

1. Navigate to the **Project Web App** (PWA) site.
2. Click on the **Settings** icon,  > **Nintex Workflow for Project Server** > **Manage Event Driven Workflows**.



3. In the Manage Event Driven Workflows Ribbon, click the **New** button to open the workflow designer.

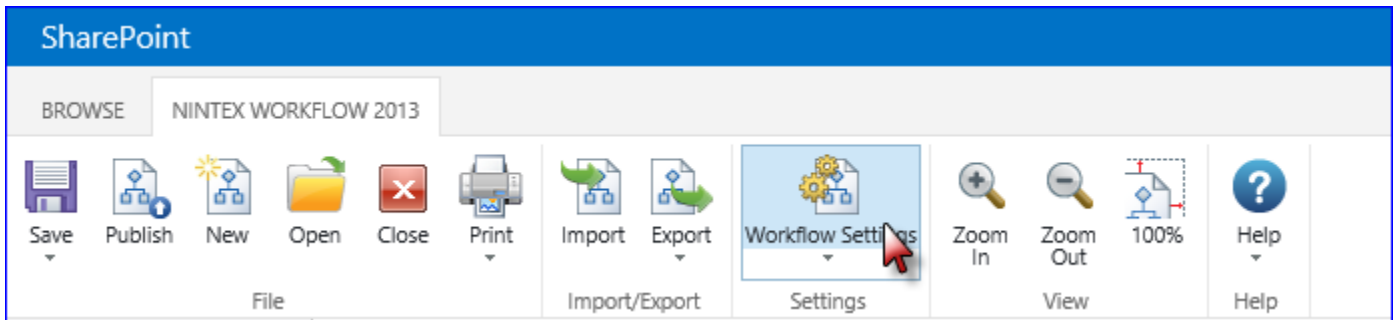
Tutorial : Event Driven Workflow



## Tutorial : Event Driven Workflow

Configure the workflow's settings and give the workflow a **Title** and **Description**.








- In the Nintex Workflow 2013 Ribbon, click on the **Workflow Settings** button.



- In the **Title**, type "Get Proposed Enterprise Resources".
- In the **Description**, type "This workflow will retrieve a list of enterprise resources with a booking type of proposed from the project plan".
- Click **Save** when completed.

### Workflow Settings □ ×

WORKFLOW SETTINGS

 Save	 Cancel	 Variables	 Association Columns	 Edit Start Form	 Manage Stage Order	 Help
Commit		Settings			Help	

#### Title and description

Title \*

Description

#### Workflow options

Publish without validation

Enable custom history messages

Expected workflow duration [Click here to configure](#)

Task list  ▼

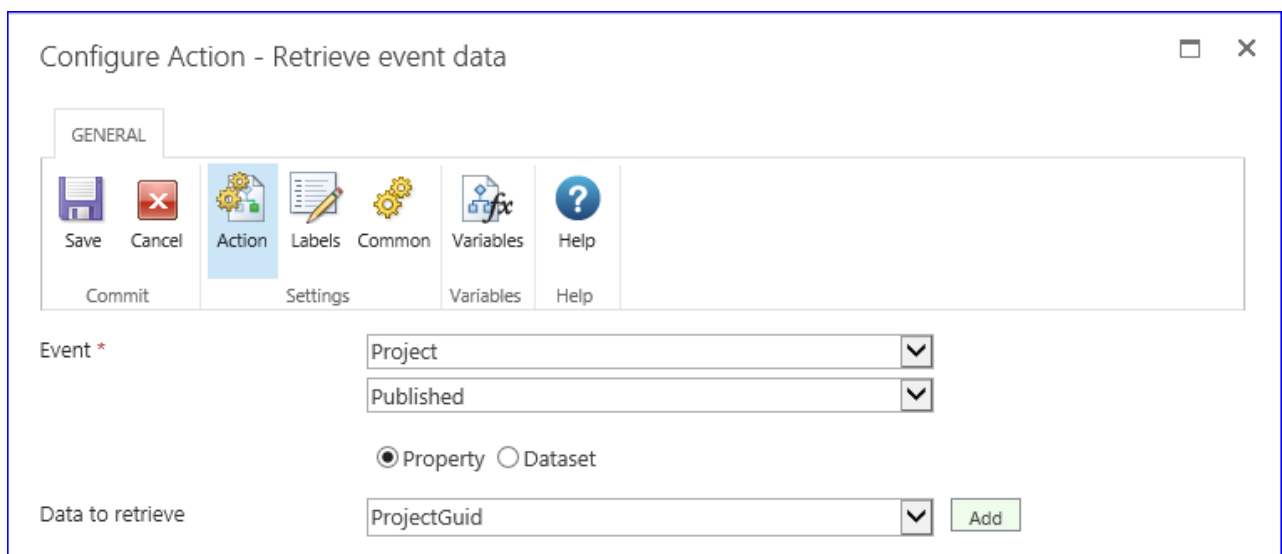
History list  ▼

## Designing the Check Baseline Workflow

The event driven workflow will be associated with the Project Published event.

In order to identify which project was published, the *Retrieve event data* action will be added to the workflow to provide that information.

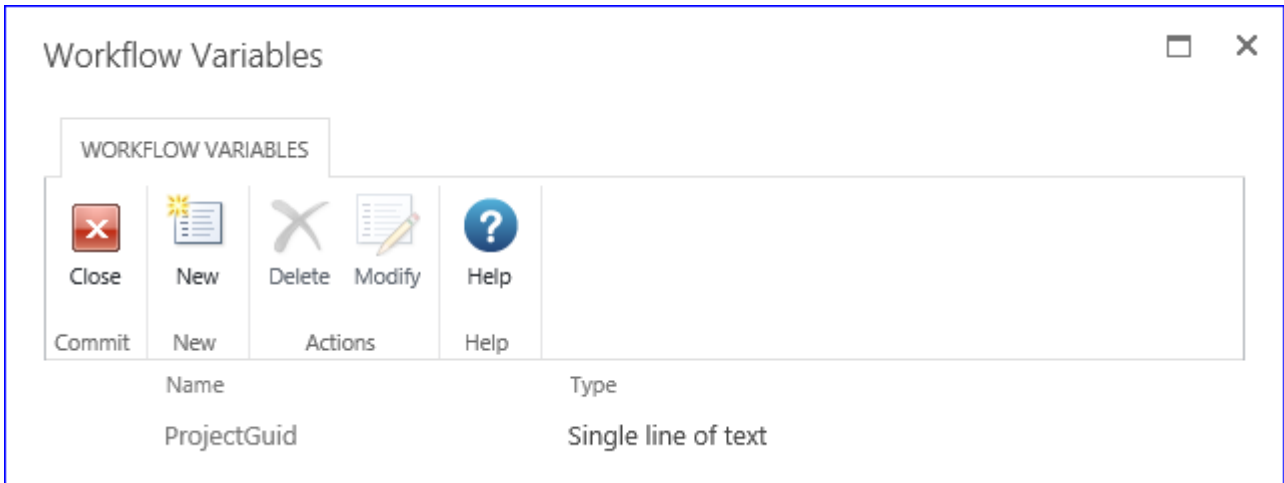
1. Navigate to the **Workflow Actions** toolbox and click the **Project Server** category to display the workflow actions specific to Project Server.
2. Drag and drop a **Retrieve event data** action onto the design canvas.
3. In the **Retrieve event data** drop-down menu, click **Configure**.
  - a. In the **Event** field, select **Project; Published** and **Property**
  - b. In the **Data to retrieve** field, select **ProjectGuid** and click **Add**.



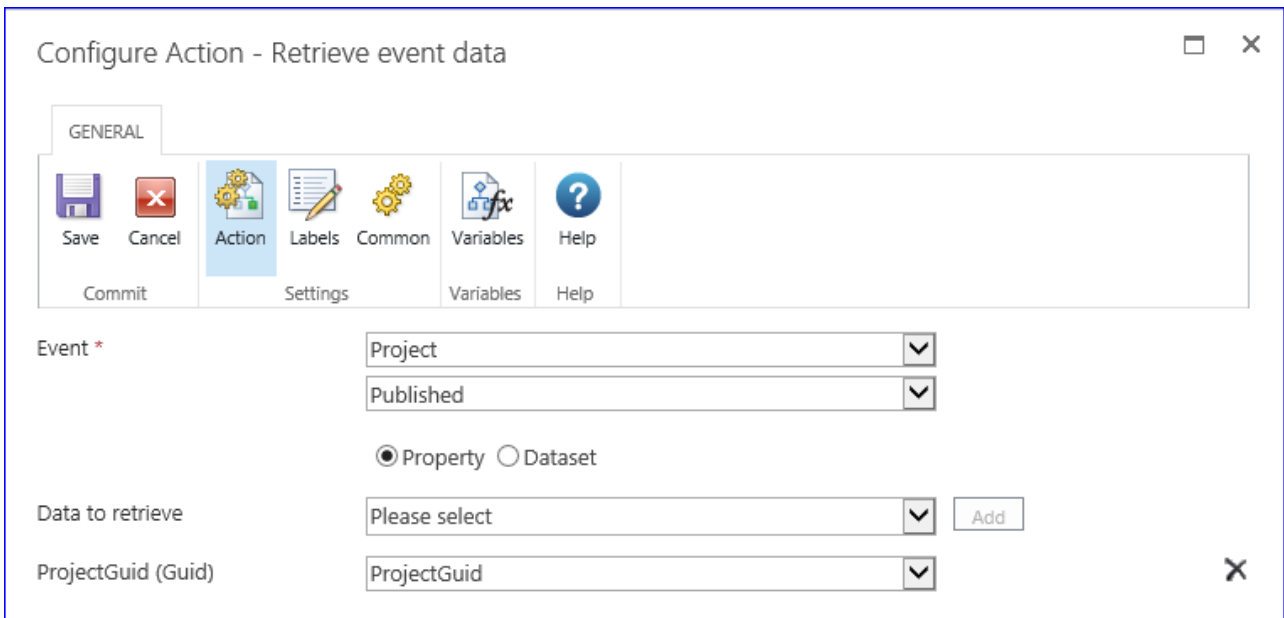
Create a workflow variable to store the ProjectGuid.

4. In the Retrieve event data Ribbon, select **Variables**.
5. In the **Workflow Variables** dialog box, click **New**.
  - a. In the **Name** box, enter "ProjectGUID".
  - b. In the **Type** field, select **Single line of text**.
  - c. Click **Save** and **Close**.





6. In the **ProjectGuid (Guid)** field, select the workflow variable **ProjectGUID**. Click **Save**.



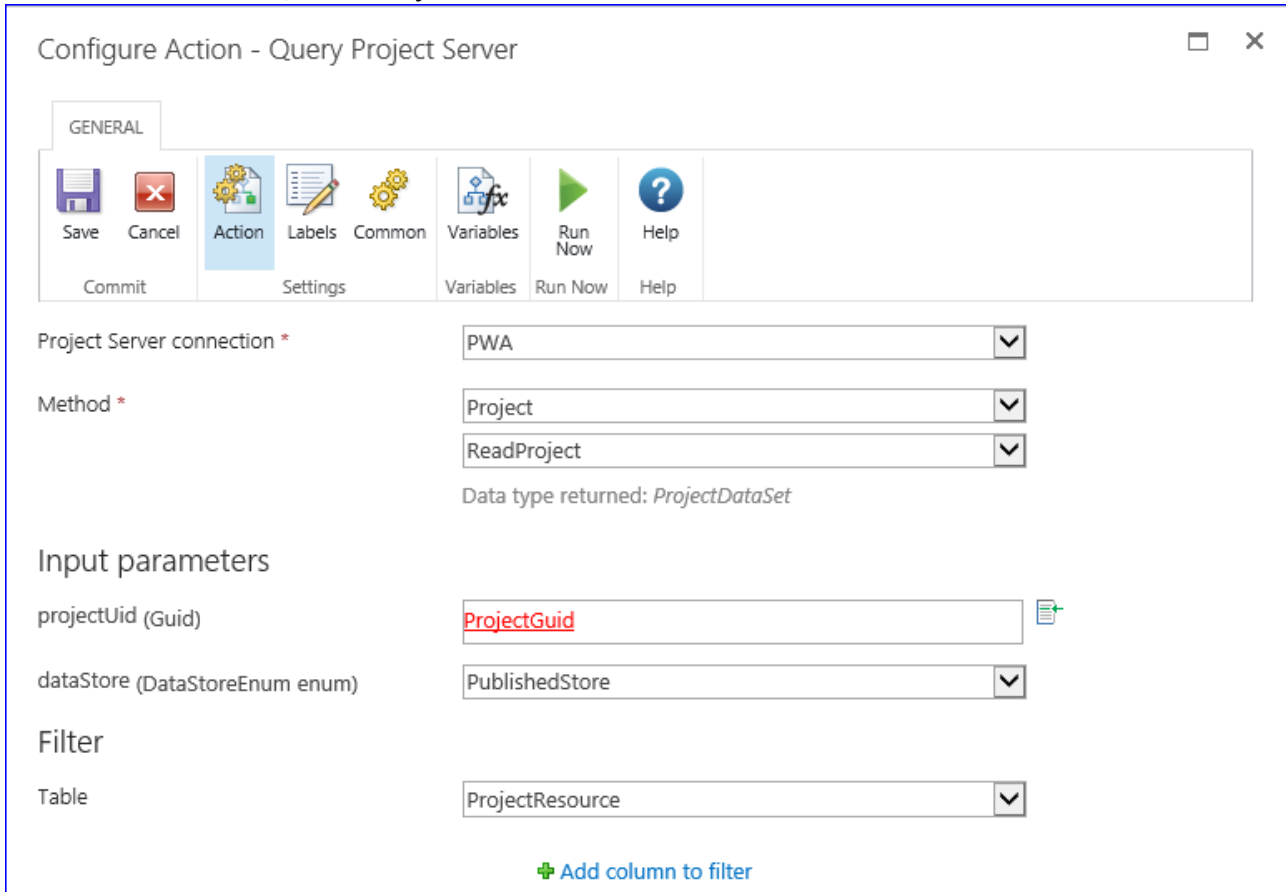
The next section describes the steps required to query the published project plan to retrieve list enterprise resources with a booking type of proposed.

7. Drag and drop a **Query Project Server** action onto the design canvas.
8. Configure the following settings:
  - a. In the **Project server connection** field, select **Configure new connection**.
  - b. Specify the connection details to your Project Web App (PWA) site in the Project Server connection fields. Click on **Verify connection**.

## Tutorial : Event Driven Workflow

Note: For more information on the Project Server connection and how to create a new Project Server connection to a PWA site, refer to the Help Files included in the product.

- c. In the **Method** field, select **Project; ReadProject**.
- d. In the **projectUid (Guid)** field, select the **Insert reference** function. In the **Workflow Variables** tab, locate the **ProjectGUID** variable. Click **OK**.
- e. In the **dataStore (DataStoreEnum enum)** field, select **PublishedStore**.
- f. In the **Table** field, select **ProjectResource**.



Configure Action - Query Project Server

GENERAL

Save Cancel Action Labels Common Variables Run Now Help

Commit Settings Variables Run Now Help

Project Server connection \* PWA

Method \* Project

ReadProject

Data type returned: *ProjectDataSet*

Input parameters

projectUid (Guid) ProjectGuid

dataStore (DataStoreEnum enum) PublishedStore

Filter

Table ProjectResource

+ Add column to filter

9. In the **Filter** section, click on **Add column to filter**.
10. Configure the following settings:
  - a. In the **Where the column** field, select **RES\_BOOKING\_TYPE** and **Equals**.
  - b. In the third **Where the column** box, type "1".
  - c. Click on **Add column to filter**.

Configure Action - Query Project Server

GENERAL

Save Cancel Action Labels Common Variables Run Now Help

Commit Settings Variables Run Now Help

Project Server connection \* PWA

Method \* Project  
ReadProject

Data type returned: ProjectDataSet

Input parameters

projectUid (Guid) ProjectGuid

dataStore (DataStoreEnum enum) PublishedStore

Filter

Table ProjectResource

Where the column RES\_BOOKING\_TYPE

Equals

1

+ Add column to filter

- d. Select **And** in the drop-down menu.
- e. In the **Where the column** field, select **RES\_IS ENTERPRISE\_RESOURCE** and **Equals**.
- f. In the third **Where the column** box, type "True".

## Tutorial : Event Driven Workflow

□ ×
Configure Action - Query Project Server

GENERAL

Save
 Cancel
 Action
 Labels
 Common

Variables
 Run Now
 Help

Commit
Settings
Variables
Run Now
Help

Project Server connection \* ▼

Method \* ▼

ReadProject ▼

Data type returned: *ProjectDataSet*

Input parameters

projectUid (Guid) 📄➕

ProjectGuid

dataStore (DataStoreEnum enum) ▼

PublishedStore

Filter

Table ▼

ProjectResource

Where the column ×

RES\_BOOKING\_TYPE ▼

Equals ▼

1 📄➕

And ▼

Where the column ×

RES\_IS\_ENTERPRISE\_RESOURCE ▼

Equals ▼

True 📄➕

➕ Add column to filter

Create a workflow variable to store the list of resources that match the query configured above.

11. Select **Variables** in the Query Project Server Ribbon and select **New**.
  - a. In the **Name** box, type "ProjectResourcesGUID".
  - b. In the **Type** field, select **Collection**. Click **Save** and **Close**.

Workflow Variables

WORKFLOW VARIABLES

Close New Delete Modify Help

Commit New Actions Help

Name	Type
ProjectGuid	Single line of text
ProjectResourcesGUID	Collection

12. In the **Select column** field, select **RES\_UID** and click **Add**.
13. In the **RES\_UID** field, select the **ProjectResourcesGUID** variable to store the list of resources that are returned by the query.
14. Click **Save**.

Tutorial : Event Driven Workflow

Configure Action - Query Project Server

GENERAL

Save Cancel Action Labels Common Variables Run Now Help

Commit Settings Variables Run Now Help

projectUid (Guid)

dataStore (DataStoreEnum enum)

Filter

Table

Where the column

And

Where the column

+ Add column to filter

Store results \*

Select column  Add

**ProjectResource**

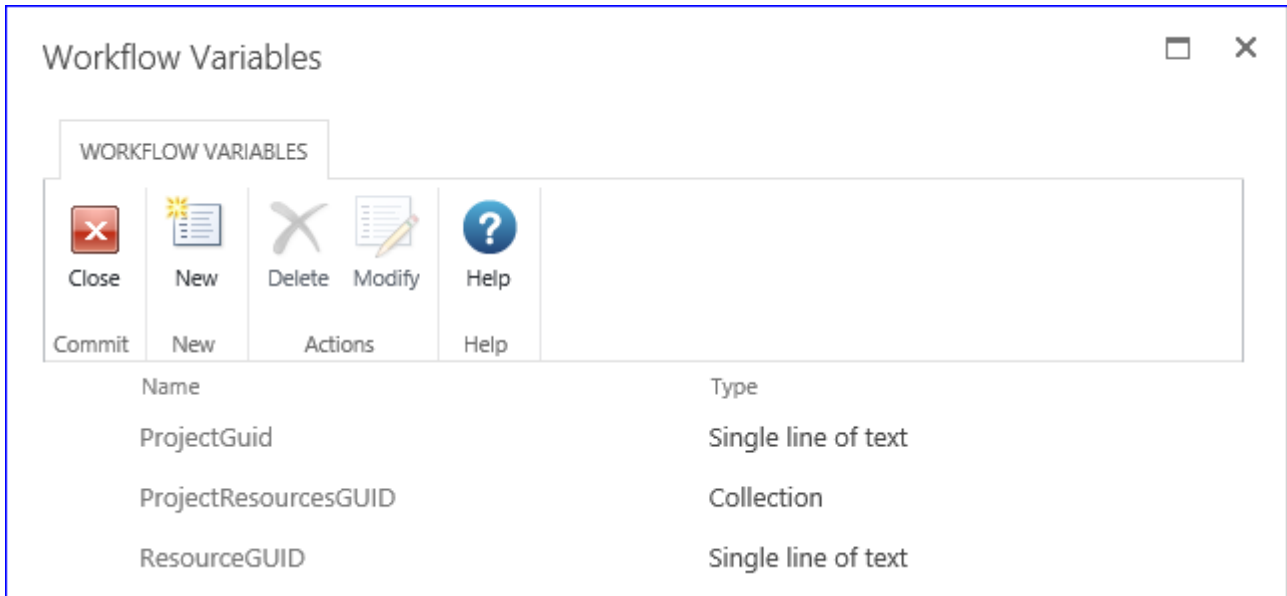
RES\_UID

+ Error handling

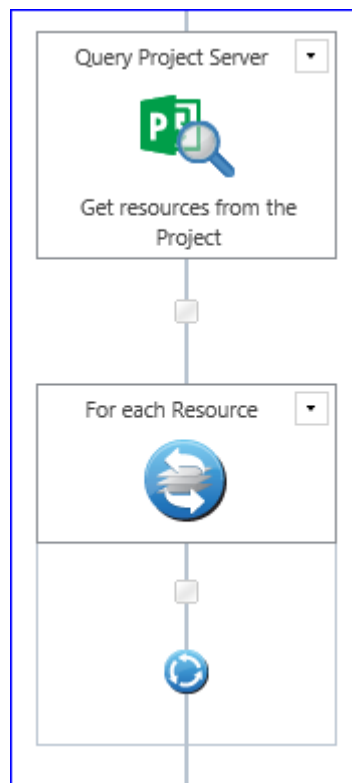
Next, retrieve and log the display name for each resource GUID that is in the ProjectResourcesGUID collection workflow variable in the workflow history list.

Create a workflow variable to store the individual Resource GUID of each resource from the ProjectResourcesGUID collection variable.

8. In the Nintex Workflow 2013 Ribbon, select the **Workflow Settings** drop-down menu and click on **Variables**.
9. In the **Workflow Variables** dialog box, click **New**.
  - a. In the **Name** box, type "ResourceGUID".
  - b. In the **Type** field, select **Single line of text**. Click **Save** and **Close**.



10. Drag and drop a **For each** action onto the design canvas.



11. Configure the following settings:

- In the **Target collection** field, select **ProjectResourcesGUID**.
- In the **Store result in** field, select **ResourcesGUID**.
- Click **Save**.

Tutorial : Event Driven Workflow

Configure Action - For each

GENERAL

Save Cancel Action Labels Common Variables Help

Commit Settings Variables Help

Target collection \* ProjectResourcesGUID

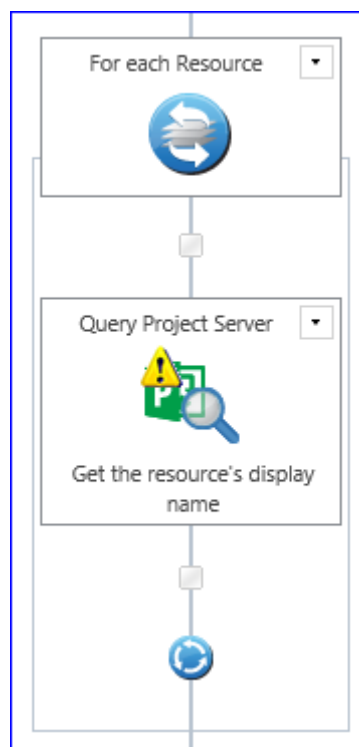
Store result in \* ResourceGUID

Index

Stop processing

*When this variable is true the loop will not process any more collection items.*

12. Drag and drop a **Query Project Server** action within the **For each** action that was just configured.



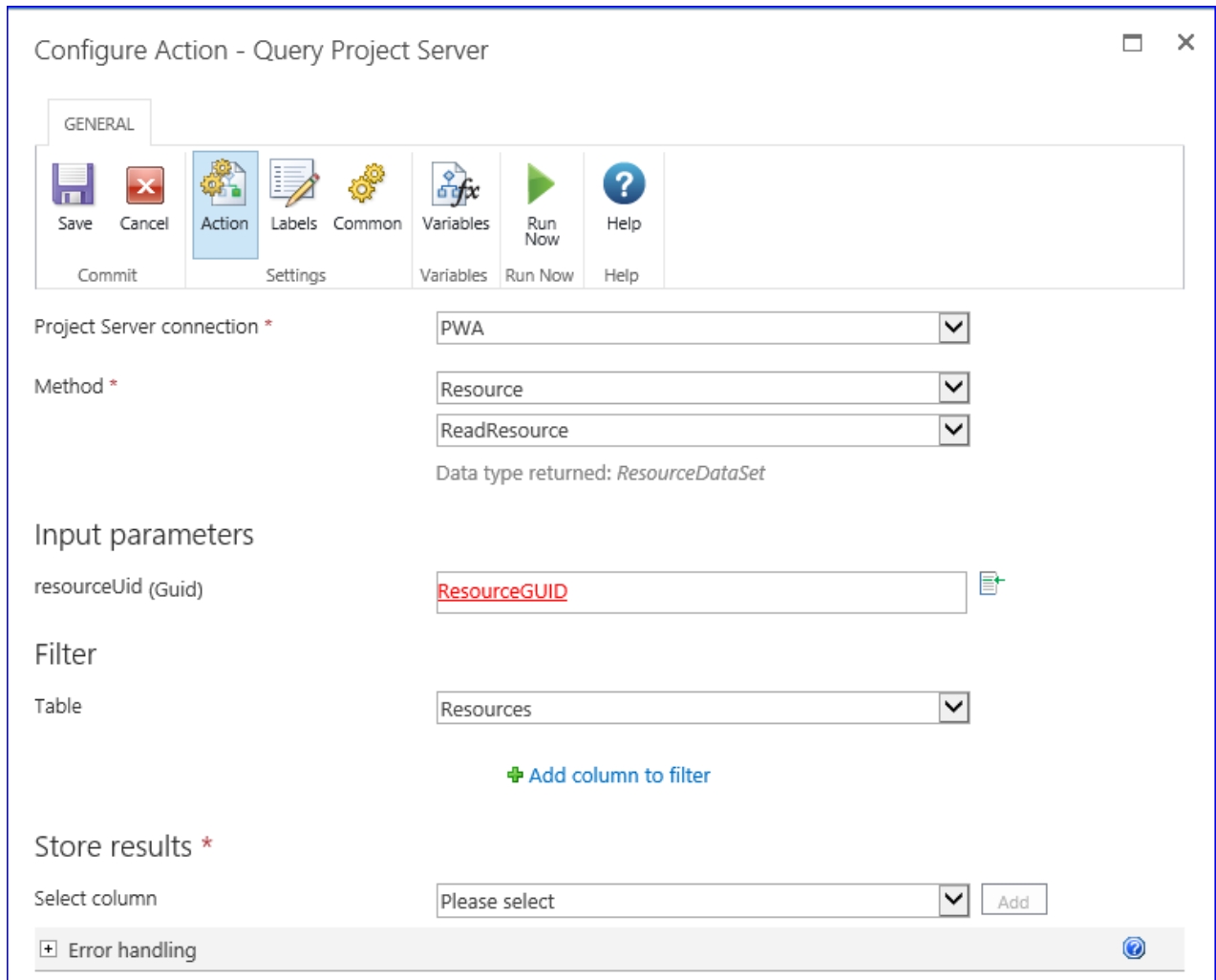
13. Configure the following settings:

- d. In the **Project Server Connection** field, select **Configure new connection**.
- e. Specify the connection details to your Project Web App (PWA) site in the Project Server connection fields. Click on **Verify connection**.



Note: For more information on the Project Server connection, refer to the Help Files included in the product.

- f. In the **Method** field, select **Resource; ReadResource**.
- g. In the **resourceUid (Guid)**, select the **Insert reference** icon. In the **Workflow Variables** tab, locate the **ResourcesGUID** variable. Click **OK**.
- h. In the **Table** field, select **Resources**.



Configure Action - Query Project Server

GENERAL

Save Cancel Action Labels Common Variables Run Now Help

Commit Settings Variables Run Now Help

Project Server connection \* PWA

Method \* Resource  
ReadResource

Data type returned: ResourceDataSet

Input parameters

resourceUid (Guid) ResourceGUID

Filter

Table Resources

+ Add column to filter

Store results \*

Select column Please select Add

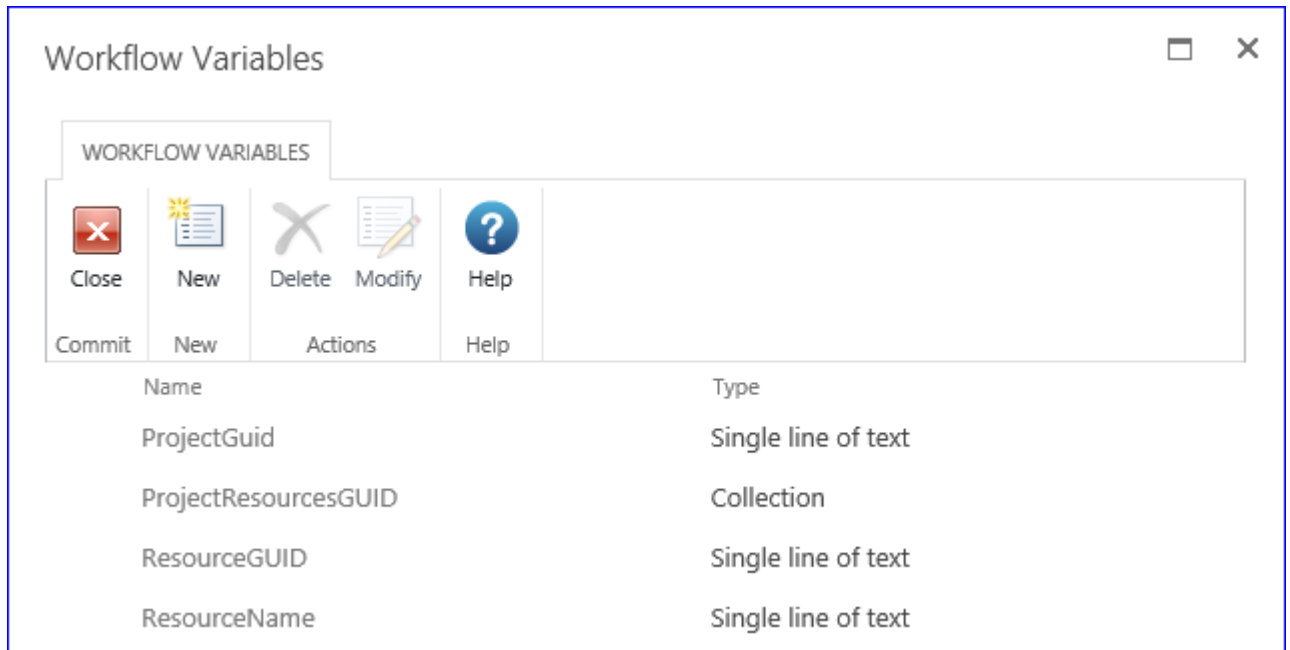
+ Error handling

Create a workflow variable to be used for storing the resource name of the query configured above.

- 14. In the Query Project Server Ribbon, select **Variables**.
- 15. In the **Workflow Variables** dialog box, click **New**.
  - a. In the **Name** box, type "ResourceName".
  - b. In the **Type** field, select **Single line of text**.

Tutorial : Event Driven Workflow

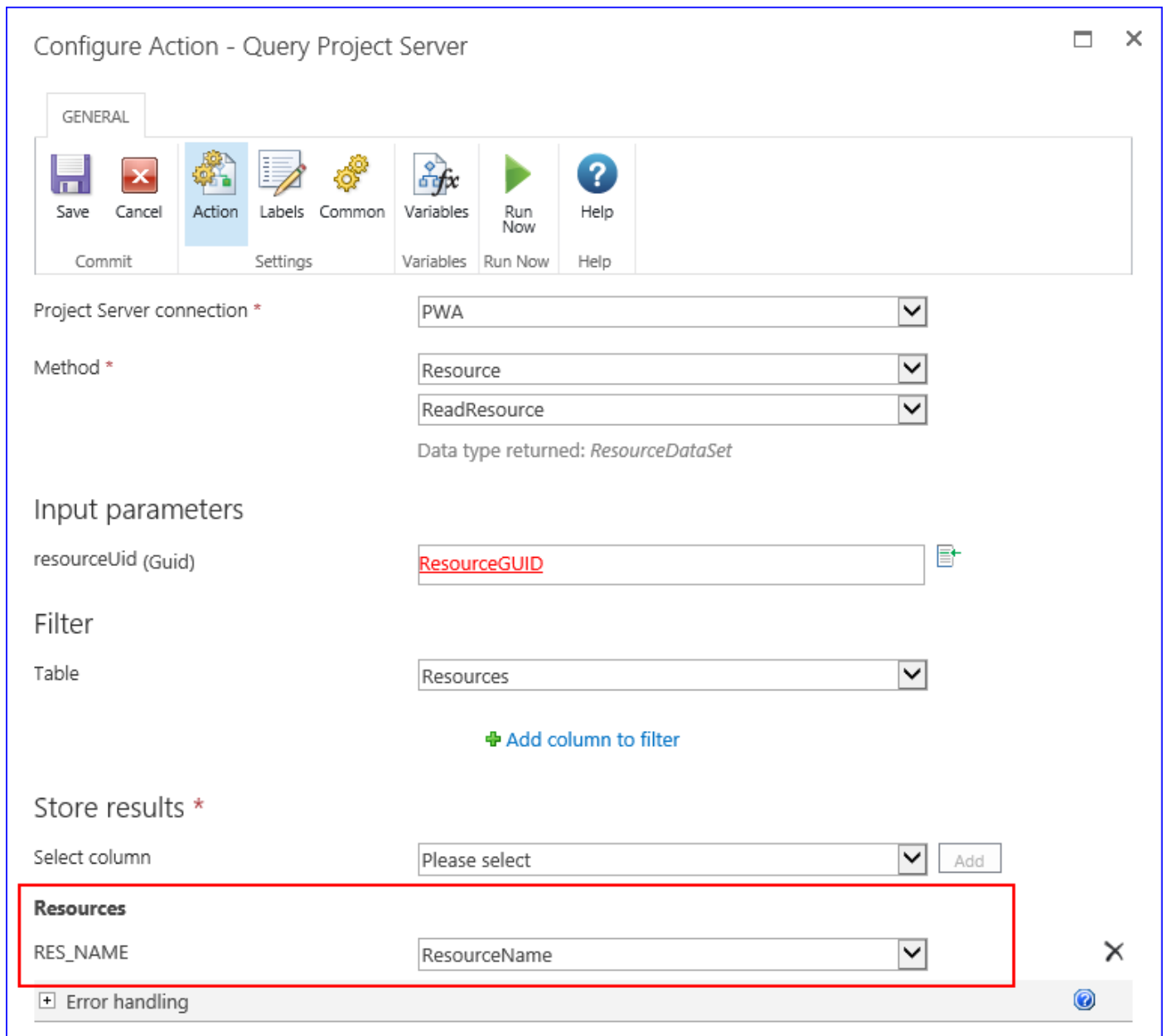
c. Click **Save** and **Close**.



16. In the **Select column**, select **RES\_NAME** and click **Add**.

17. In the **RES\_NAME** field, select the **ResourceName** variable. This will store the name of the resource that is returned by the query.

18. Click **Save**.



Configure Action - Query Project Server

GENERAL

Save Cancel Action Labels Common Variables Run Now Help

Commit Settings Variables Run Now Help

Project Server connection \* PWA

Method \* Resource

ReadResource

Data type returned: ResourceDataSet

Input parameters

resourceUid (Guid) ResourceGUID

Filter

Table Resources

+ Add column to filter

Store results \*

Select column Please select Add

**Resources**

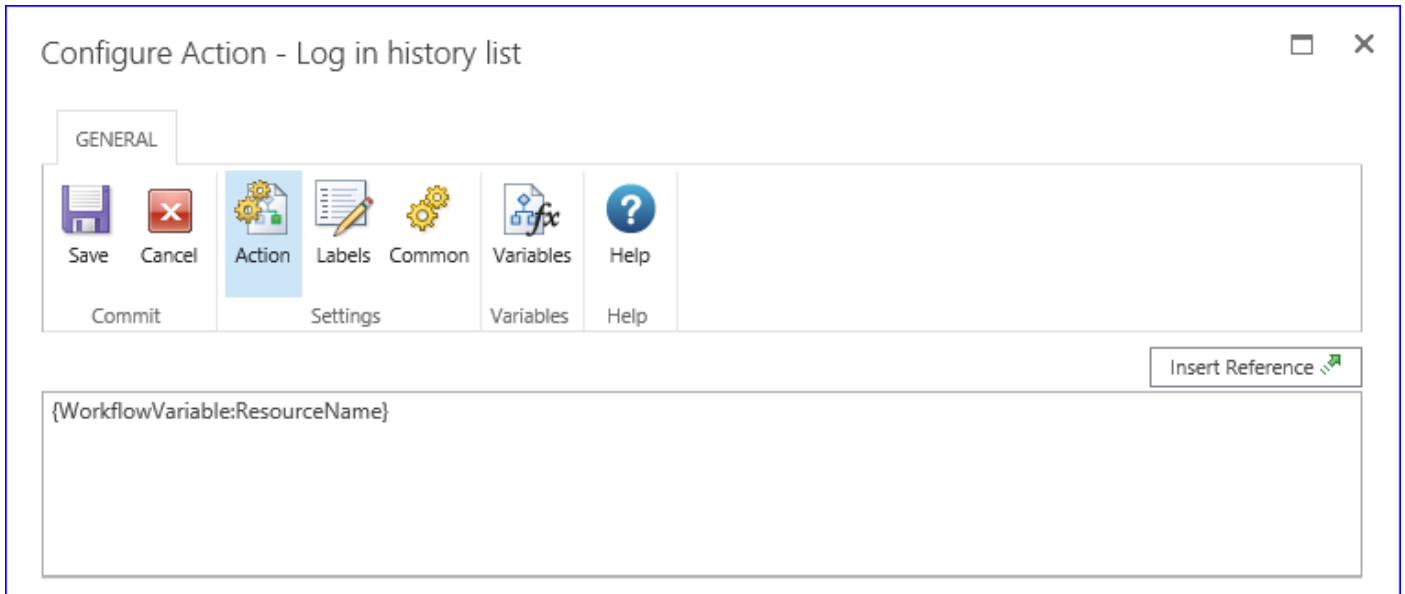
RES\_NAME ResourceName

+ Error handling

The next step is to log the Resource Name in the workflow history list.

19. Drag and drop a **Log in history list** action within the **For each** action below the **Query Project Server** action.
20. Configure the following settings:
  - a. Click the **Insert Reference** icon.
  - b. In the **Workflow Variables** tab, locate the **ResourceName** variable.
  - c. Click **OK** and **Save**.

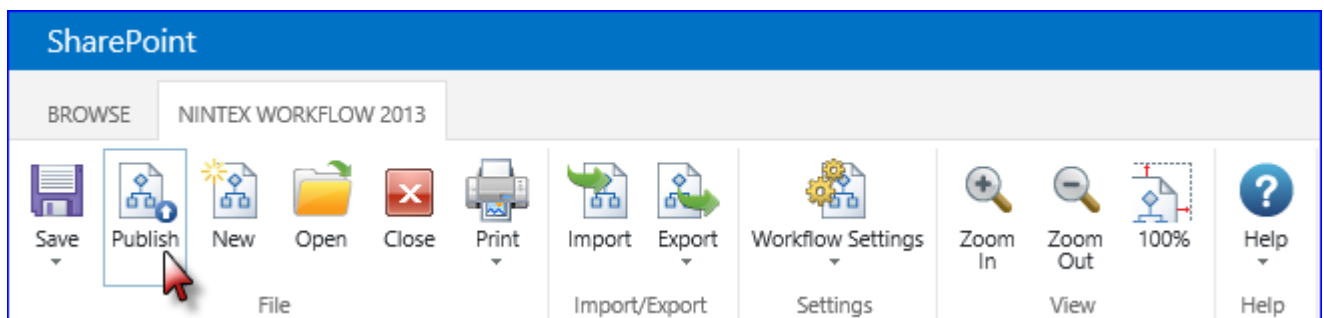
Tutorial : Event Driven Workflow



**Publish the workflow**

The capabilities available will depend on the version of Nintex Workflow 2013 which is installed and licensed on the farm.

21. In the Nintex Workflow 2013 Ribbon, click **Publish**.




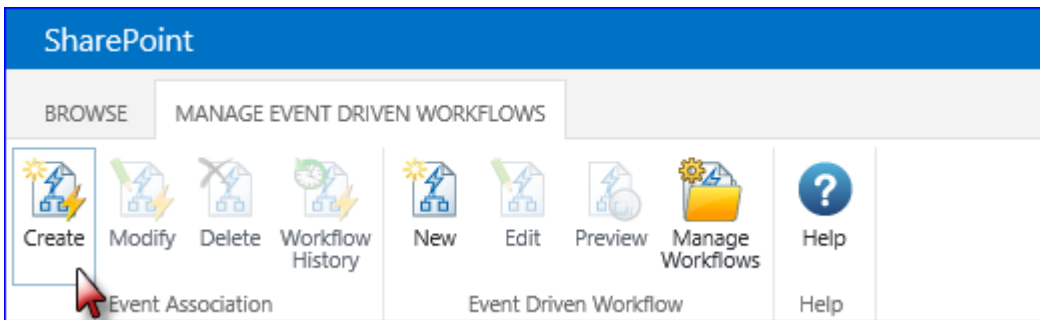
22. In the **Set Title & Description** dialog, click **Submit** and then **OK**.

23. In the Nintex Workflow 2013 Ribbon, click **Close**.

## Associating the Event Driven Workflow with a Microsoft Project Server event

The event driven workflow created in the section above must now be associated to the Project-Published event so that the workflow is automatically started when a Project-Published event occurs in Microsoft Project Server.




1. Navigate to the Project Web App (PWA) site.
2. Select the **Settings** icon,  > **Nintex Workflow for Project Server** > **Manage Event Driven Workflows**.
3. In the **Manage Event Driven Workflows** Ribbon, click **Create**.






4. In the **Associate a workflow to an event** dialog, configure the following settings:
  - a. In the **Event** field, select **Project; Published**.
  - b. In the **Workflow** field, select **Get Proposed Enterprise Resources**.
  - c. In the **Description** field, type "Get proposed enterprise resources in the project plan".
  - d. Click **Save** when completed.

### Associate a workflow to an event ✕

EVENT DRIVEN WORKFLOW ASSOCIATION

 Save
 Cancel
 Help

 Commit
 Help

 Adding a new event driven workflow association may require a few minutes to take effect.

Event \* Project ▼

Published ▼

Workflow \* Get Proposed Enterprise Resources ▼


Description

Get proposed enterprise resources in the project plan.

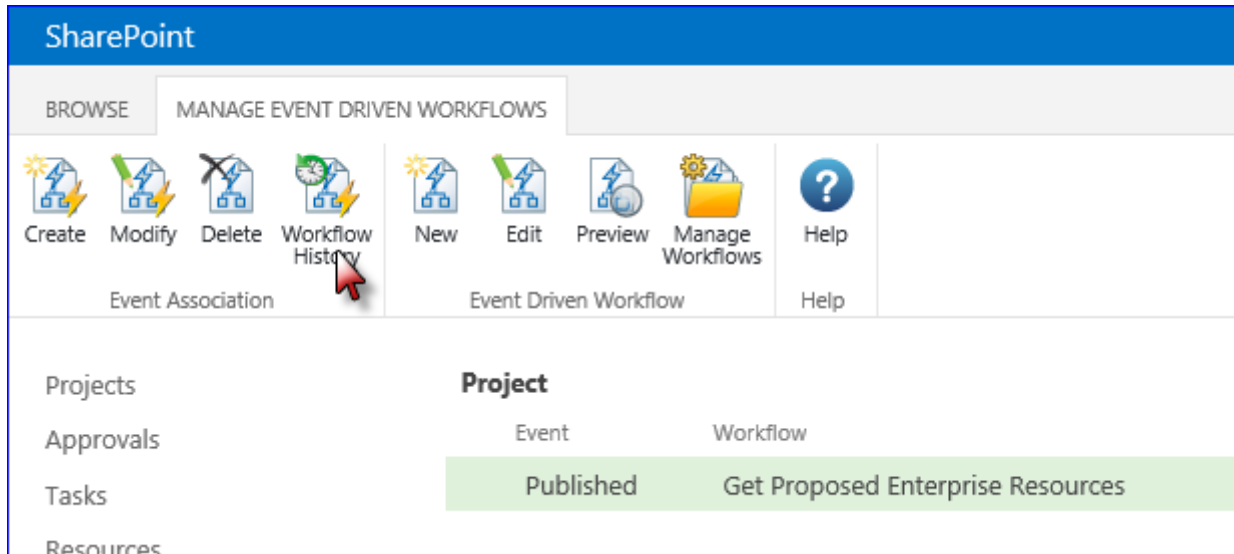
Enabled

## Next Steps

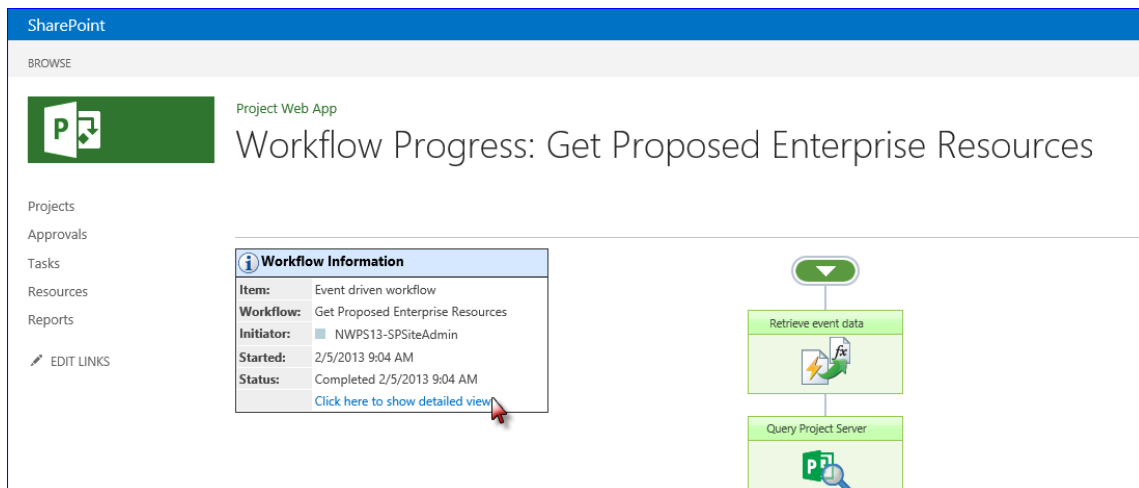
Quick steps to run the workflow:

1. Create a project plan and build the team for the project using enterprise resources.
2. Ensure that the resources' Booking Type is set to Proposed.
3. Publish the project plan to Microsoft Project Server. (Note: The workflow will start when the publish job for this project has been completed.)
4. Open the **Workflow History** of the event driven workflow association:
  - a. Navigate to the **Project Web App (PWA)** site. Click on the **Settings** icon,  > **Nintex workflow for Project Server > Manage Event Driven Workflows**.
  - b. In the **Project** section, select the **Get Proposed Enterprise Resources** event from the list.
  - c. In the Manage Event Driven Workflows Ribbon, select **Workflow History**.

Tutorial : Event Driven Workflow



- In the **Completed Workflows** section, click the link of the workflow that was most recently completed.
- In the **Workflow Information** box, select **Click here to show detailed view**.



- The list of the proposed enterprise resources will be displayed in the **Workflow Messages** section.