NINTEX Workflow 2013 for Project Server

Tutorial : Check Project Baseline Workflow

NINTEX Workflow for Everyone™

Microsoft Partner
Gold Independent Software Vendor (ISV)

Contents

| **www.nintex.com** | **support@nintex.com** |

About this tutorial

This tutorial will describe how to incorporate a check into a demand management workflow to ensure that tasks within a project are baselined before the project is allowed to move on to the next stage.

After completion of this tutorial, you will have a basic understanding of the 'Query Project Server' action and how other queries could be created to ensure project plan governance can be incorporated into demand management workflows.

This tutorial assumes you have basic knowledge in designing demand management workflows in Project Server using Nintex Workflow for Project Server 2013.

Before you start

In order to complete this tutorial, it is necessary to configure a number of stages and an enterprise custom field within the Project Web App. The steps to complete this are outlined in Appendix A.

Getting Started

Open Nintex Workflow for Project Server 2013 workflow designer.

1. Navigate to the **Project Web App** (PWA) site.

2. Click on the **Settings Icon** ⚙ > **Project Server Workflow** > **Create Demand Management Workflow**.
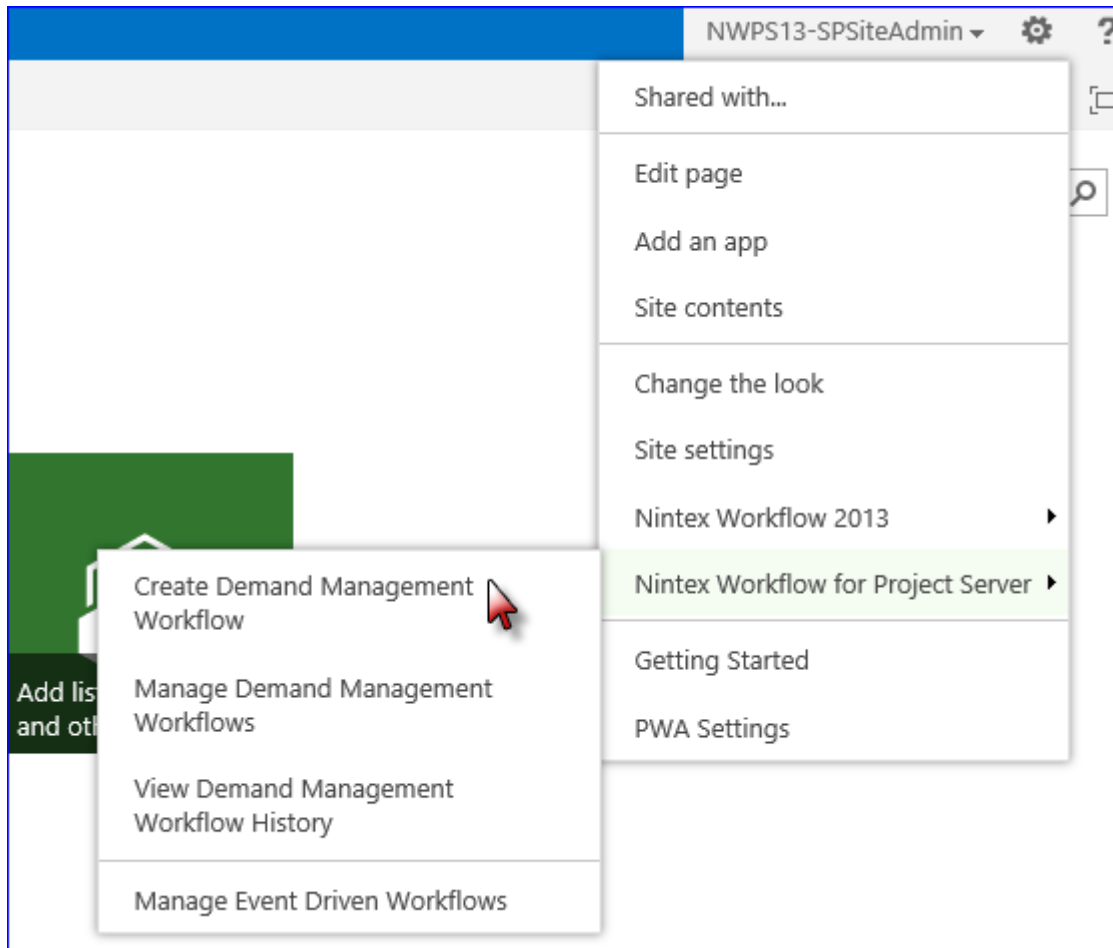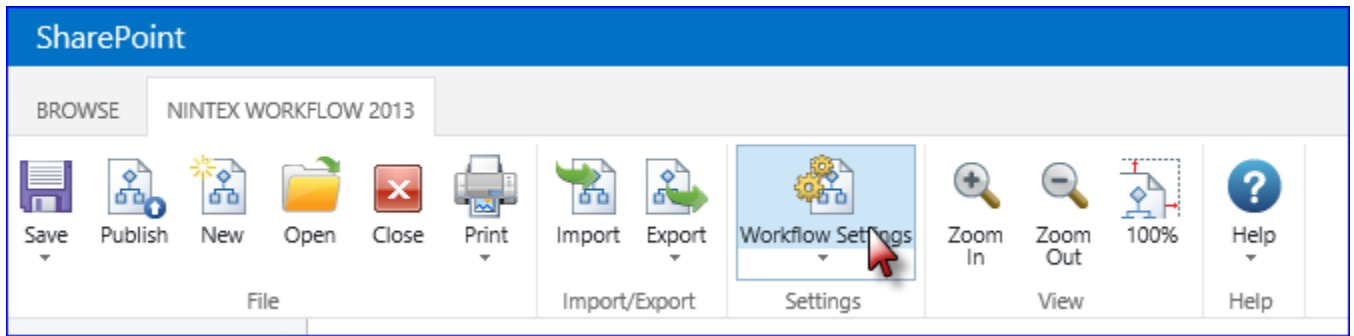

Configure the workflow's settings and give the workflow a Title and Description.

3. Select a **Blank** Workflow Template and click **Create.**

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

5. In the **Title** box, type "Check Project Baseline"**.**
6. In the **Description** box, type "This workflow will check if the tasks in the project plan have been baselined before allowing it to move to the next project stage".
7. Click **Save** when completed.

## Designing the Check Project Baseline Workflow

The initial steps within this workflow will be to set the project stage and initial status of the project to wait for an input once the workflow starts.

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 **Set project stage** workflow action onto the design canvas.
3. In the **Set project stage** drop-down menu, click **Configure**.
4. Configure the following setting:
   a. In the **Stage** field, select **Initial Proposal Details**.
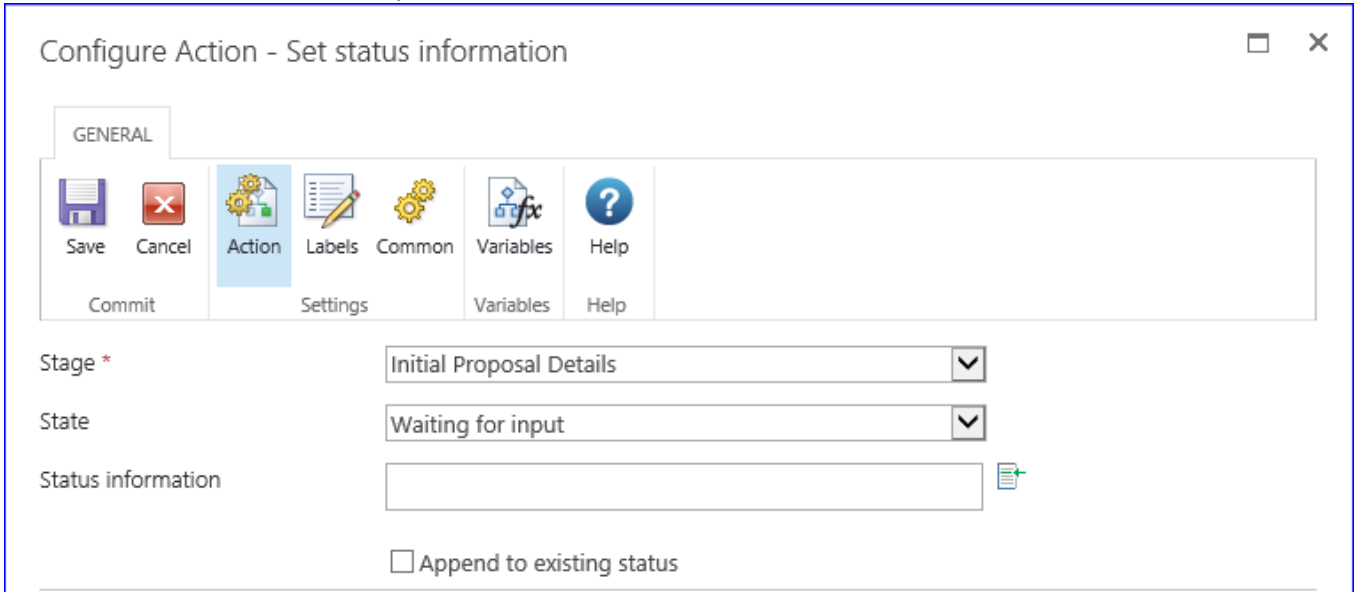   b. Click **Save.**
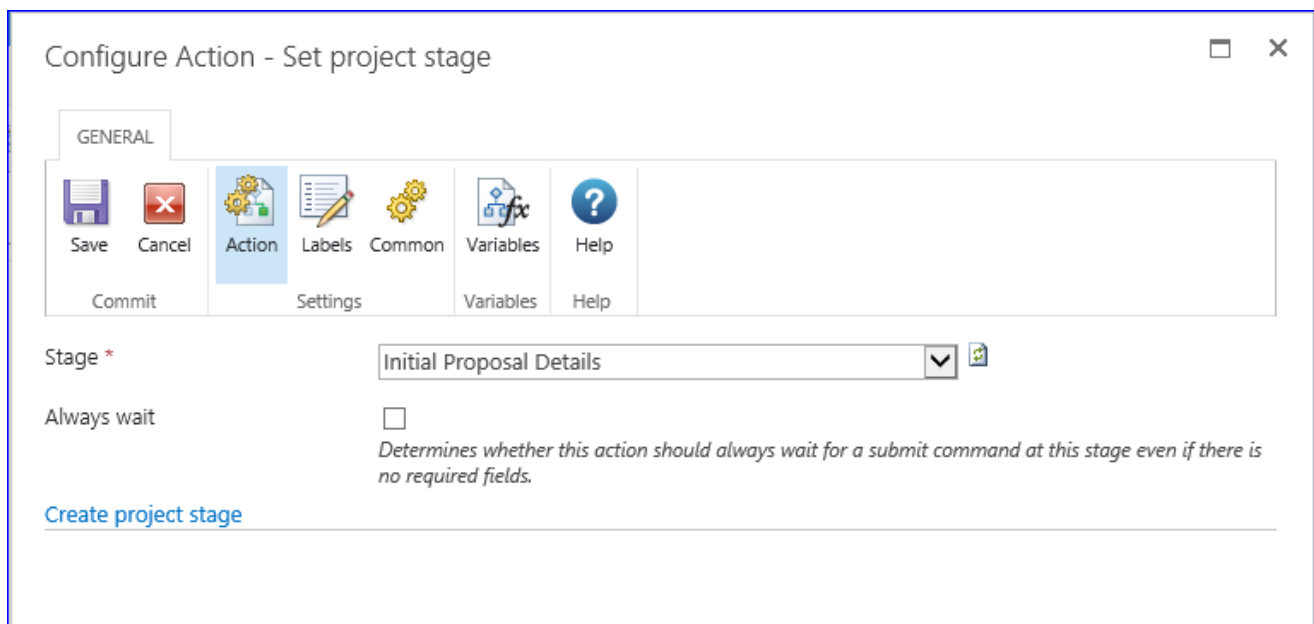
5. Drag and drop a **Set status information** workflow action onto the design canvas below the **Set project stage** action.
6. In the **Set status information** the drop-down menu, click **Configure.**

7. Configure the following settings:

a. In the **Stage** field, select **Initial Proposal Details**.

b. In the **State** field, select **Waiting for input**.

c. Click **Save** when completed.

7. Drag and drop another **Set project stage** action onto the design canvas and place below the **Set status information** action.

8. In the **Set project stage** drop-down menu, click **Configure.**

a. In the **Stage** field, select **Scheduling**.

b. Click **Save.**

The next steps are to check if all tasks in the project plan have been baselined when the project is submitted. A basic "loop-while" logic will be implemented to keep the project in the current stage if there are any tasks which have not been baselined.

First, create a workflow variable to be used for storing the condition of the loop. This workflow variable will ensure that the project remains in the current stage until the condition is no longer valid.

9. In the Nintex Workflow 2013 Ribbon, select the **Workflow Settings** menu and click on **Variables.**
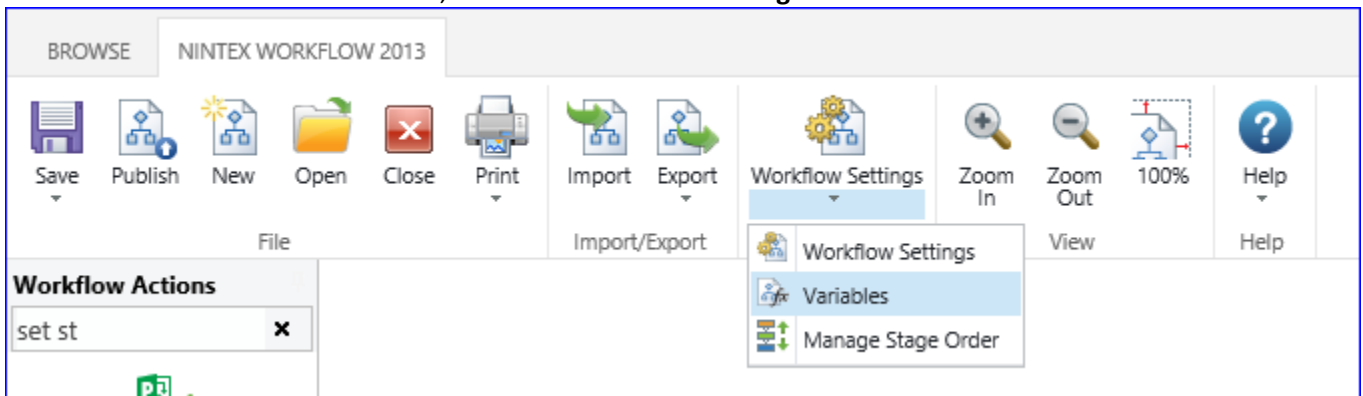
10. In the **Workflow Variables** dialog box, click **New.**
    a. In the **Name** box, type "All Task Baselined".
    b. In the **Type** field, select **Yes/No** and click **Save**.
    c. Click **Close**.
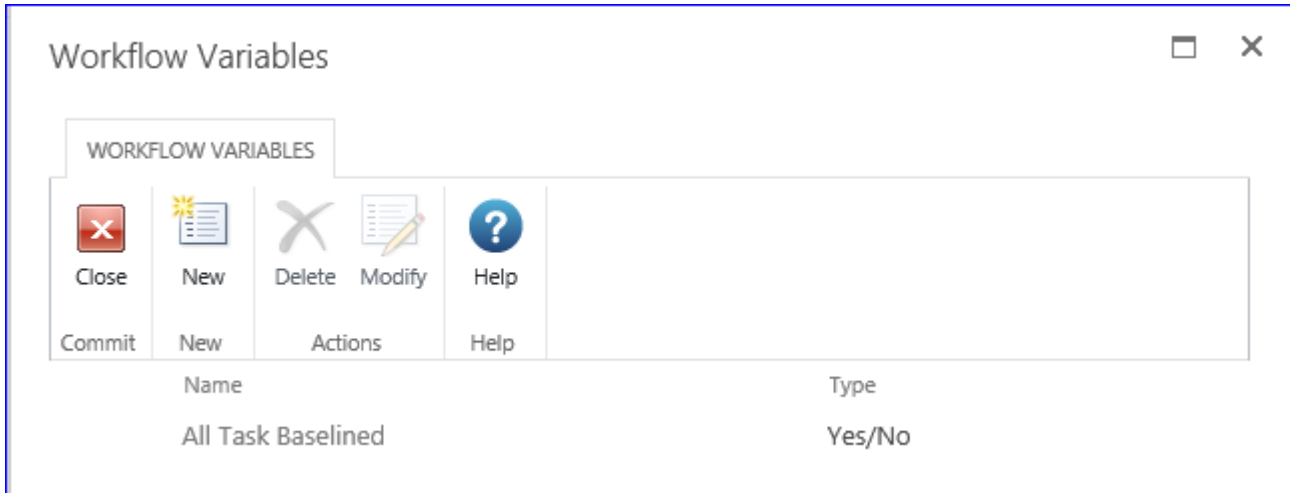
Add and create the Loop logic within the workflow.

11. Drag and drop a **Loop** action and place below the last action.
12. Configure the following settings:
    a. In the **Condition** field, select **If any value equals value.**
    b. In the **Where** field, select **Workflow Data**; **All Task Baselined**.; **equals**; **Value**; and **No.**
    c. Click **Save.**
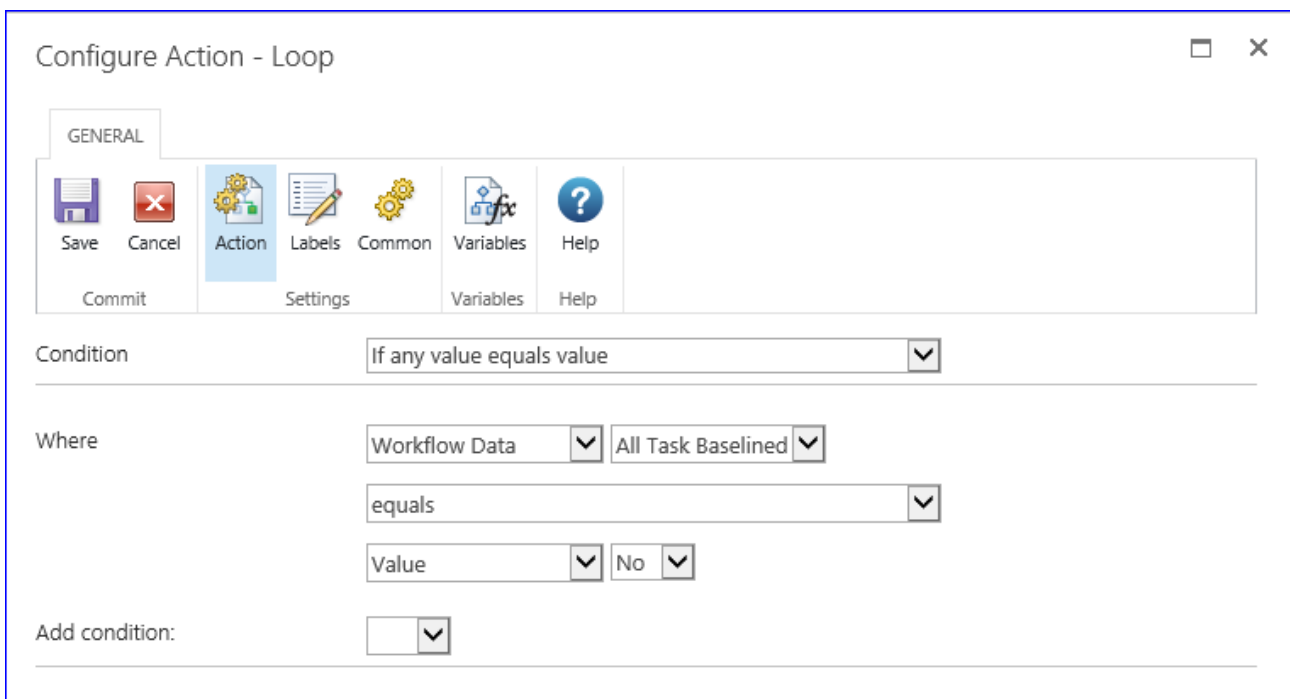
13. Drag and drop a **Wait for submit** action into the **Loop** action that was just configured.

   Note: The Wait for submit action does not need to be configured.

The next section will describe the steps required to query the project plan to retrieve a list of tasks that have not been baselined.

14. Drag and drop a **Query Project Server** action inside the **Loop** action and place below the **Wait for submit** action.
15. 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**.

   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 **Common** tab, locate the **Project UID**. Click **OK.**
   e. In the **dataStore (DataStoreEnum enum)** field, select **PublishedStore**.

   f. In the **Table** field, select **Task**.

Configure Action - Query Project Server

GENERAL

Save | Cancel | Action | Labels | Common | Variables | Run Now | Help

Commit | Settings | Variables | Run Now | Help

Project Server connection * | Configure new connection...

Project Server URL * | http://nwps13-1/PWA
Username * | nintextest\nwps13-spsiteadmin
Password | ••••••••••••

● Windows authentication   ○ Forms authentication

Verify connection

Method * | Project
ReadProject

Data type returned: *ProjectDataSet*

Input parameters

projectUid (Guid) | ProjectUID

dataStore (DataStoreEnum enum) | PublishedStore

Filter

Table | Task

➕ Add column to filter

Store results *

Select column | Please select | Add

16. In the **Filter** section, click on **Add column to filter**.
17. Configure the following setting:
    a. In the **Where the column** section, select **TB_START** and **Is null**.

Note: In this example, it is assumed that the project's Task Baseline Start is the only check required. There are other task baseline fields such as TB_FINISH, TB_COST, TB_WORK, TB_DUR which can be used in the filter to ensure a more thorough check is done for the baselining of tasks.

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

18. In the Query Project Server Ribbon, select **Variables.**
19. In the **Workflow Variables** dialog box, click **New**.
    a. In the **Name** box, type "ProjectTasksUID".
    b. In the **Type** field, select **Collection.**
    c. Click **Save** and **Close.**

20. In the **Select column** field, select **TASK_UID** and click **Add**.
21. In the **TASK_UID** field, select the workflow variable **ProjectTasksUID.** This will store the lists of tasks that are returned based on the query.

22. Click **Save.**

The **Query Project Server** action has now been configured to return tasks from the project if they have not been baselined.

The next step is to count the number of tasks that were returned from the query and if the number of tasks returned is 0, exit the loop.
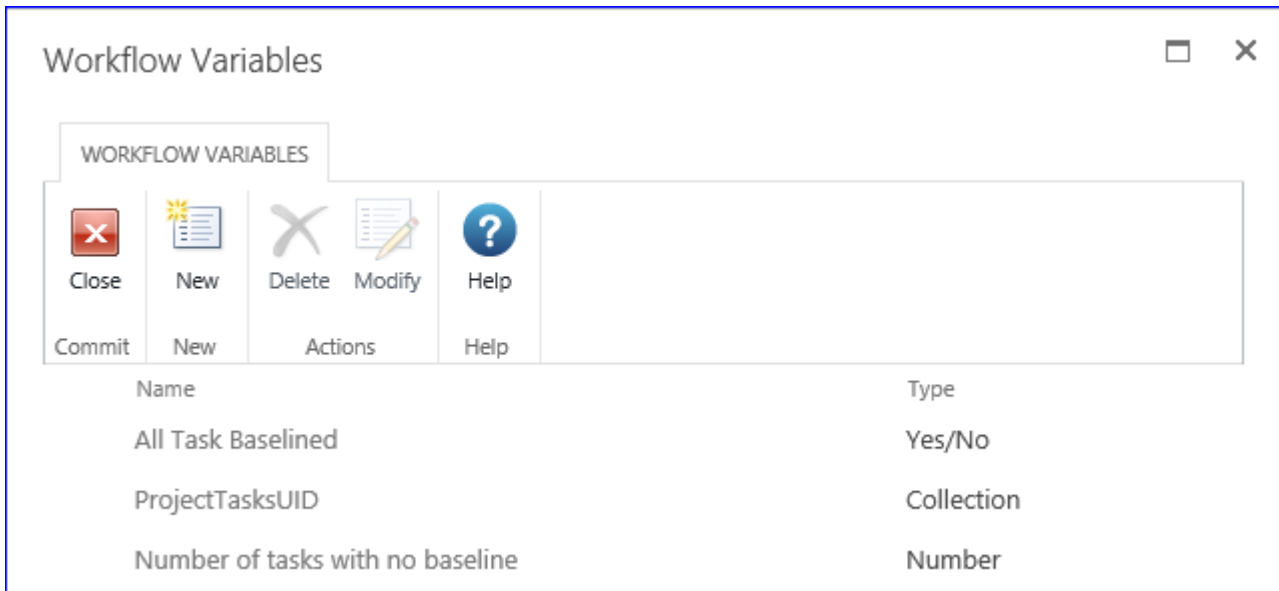
Create a workflow variable to be used for storing the total number of tasks with no baseline.

23. In the Nintex Workflow 2013 Ribbon, select the **Workflow Settings** menu and click on **Variables.**
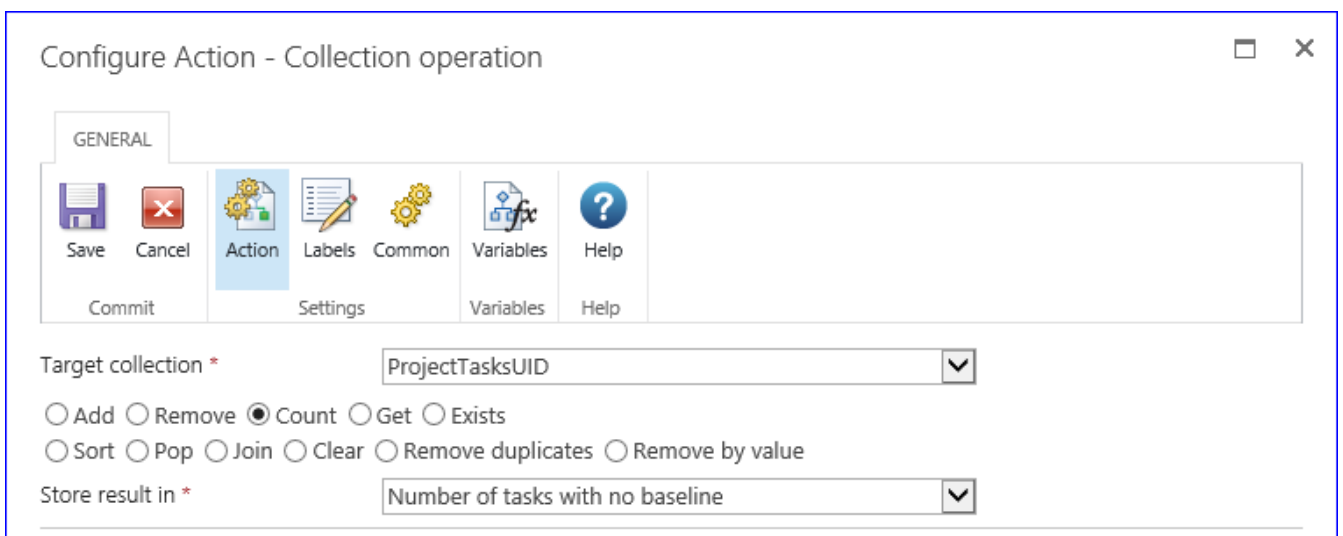24. In the **Workflow Variables** dialog box, click **New.**
    a. In the **Name** box, type "Number of tasks with no Baseline".
    b. In the **Type** field, select **Number.**
    c. Click **Save** and **Close.**

Workflow Variables

WORKFLOW VARIABLES

Close	New	Delete	Modify	Help
Commit	New	Actions	Help

Name	Type
All Task Baselined	Yes/No
ProjectTasksUID	Collection
Number of tasks with no baseline	Number

25. Drag and drop a **Collection operation** action inside the **Loop** action and below the Query Project Server action.
26. Configure the following settings:
    a. In the **Target collection** field, select **ProjectTasksUID** and **Count.**
    b. In the **Store results in** field, select **Number of Tasks with no Baseline.**
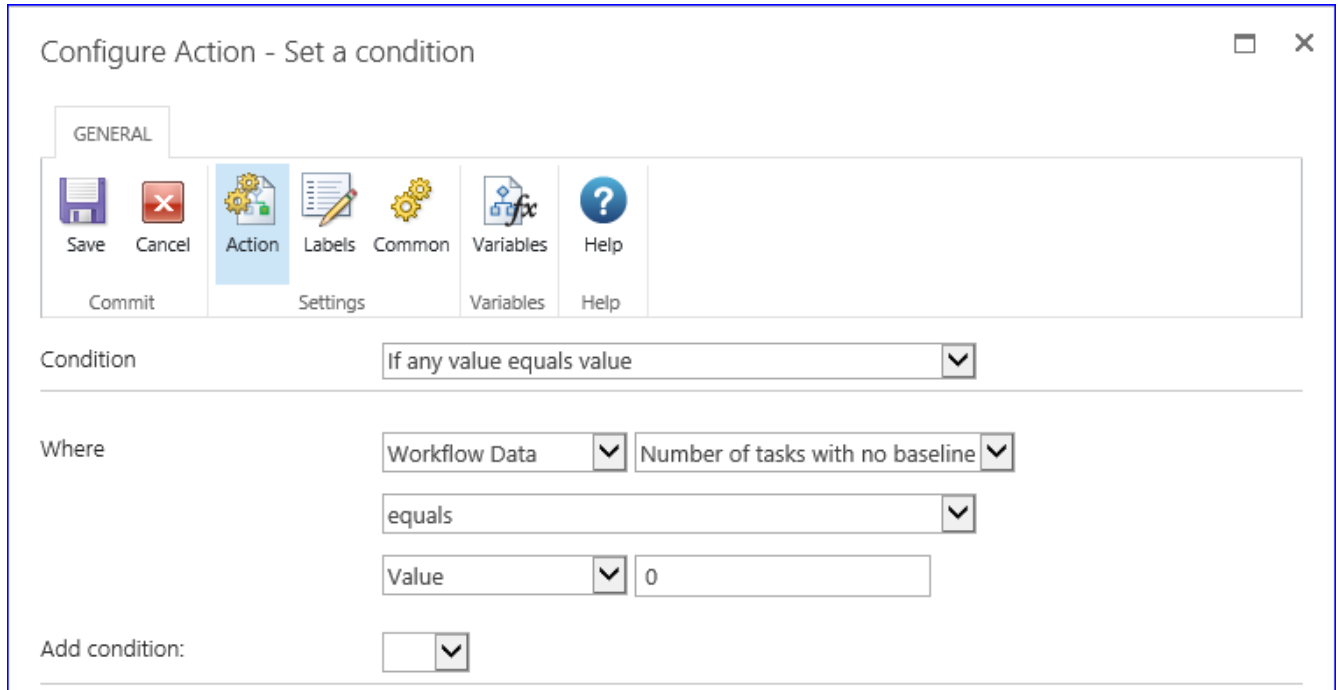   c. Click **Save.**

Configure Action - Collection operation

GENERAL

Save	Cancel	Action	Labels	Common	Variables	Help
Commit		Settings			Variables	Help

Target collection *   ProjectTasksUID

○ Add ○ Remove ● Count ○ Get ○ Exists
○ Sort ○ Pop ○ Join ○ Clear ○ Remove duplicates ○ Remove by value

Store result in *   Number of tasks with no baseline

27. Drag and drop a **Set a condition** action into the **Loop** action and below the **Collection Operation** action.

28. Configure the following settings:
    a. In the **Condition** field**,** select **If any value equals value.**
    b. In the **Where** field, select **Workflow Data**; **Number of Tasks with No Baseline**; **equals**; **Value** and **0.**
    c. Click **Save.**

If the *Number of Tasks with No Baseline* is not "0", there are still tasks in the project plan that are not baselined, the project will not move to the next stage and will wait for the project to be submitted again before performing another check.

Configure the No branch of the Set a condition workflow action.
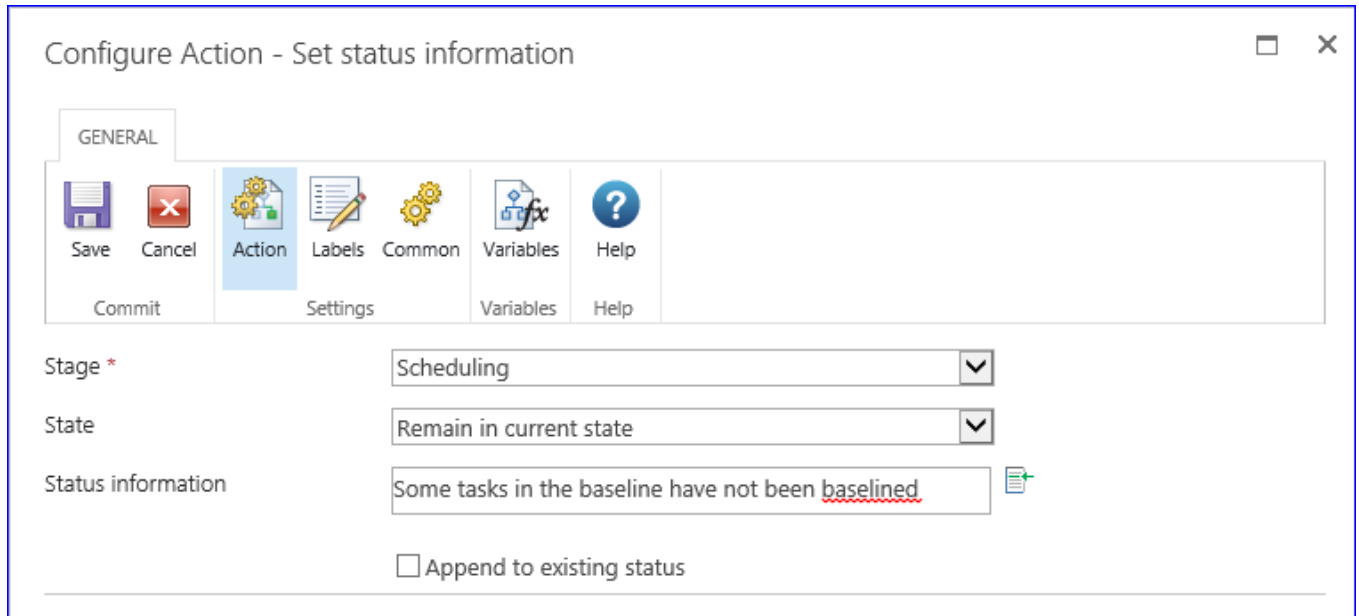
29. Drag and drop a **Set status information** action onto the **No** branch of the **Set a condition** action.
30. Configure the following settings:
    a. In the **Stage** field, select **Scheduling**.
    b. In the **State** field, select **Remain in current state**.
    c. In the **Status information** field, type "Some tasks in the project plan have not been baselined".

    d. Click **Save.**

If the *Number of Tasks with No Baseline* is "0", all tasks in the project plan have been baselined, the workflow will exit out of the loop and move to the next stage.

Configure the Yes branch of the Set a condition workflow action

31. Drag and drop a **Set variable** action onto the **Yes** branch of the **Set a condition** action below the **Set status information** action.
32. Configure the following settings:
    a. In the **Set** field, select **All Task Baseline**.
    b. In the **Equals** field, select **Value; Yes**.

33. Drag and drop a **Set status information** action onto the **Yes** branch of the **Set a condition** action.
34. Configure the following settings:
    a. In the **Stage** field, select **Scheduling**.
    b. In the **State** field, select **Remain in current state**.
    c. In the **Status information** field, type "All tasks in the project plan have been baselined".
    d. Click **OK** and then **Save.**

Configure Action - Set status information

GENERAL

Save | Cancel | Action | Labels | Common | Variables | Help

Commit | Settings | Variables | Help
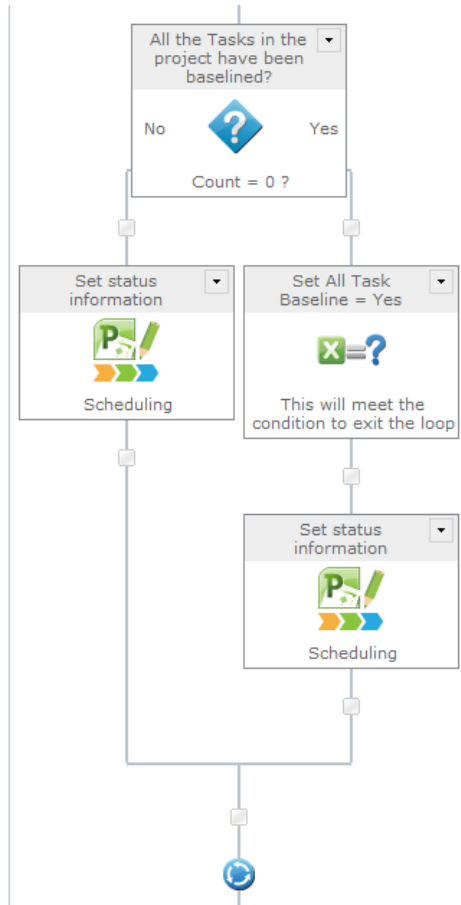
Stage *	Scheduling
State	Remain in current state
Status information	All tasks in the baseline have been baselined
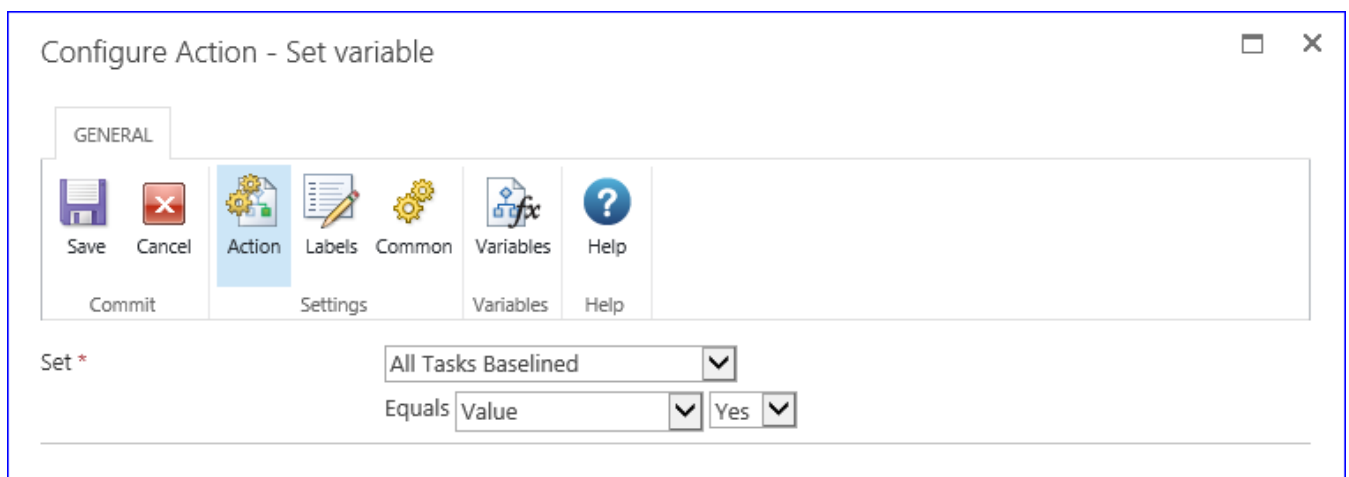
☐ Append to existing status

The Set condition logic is now configured and should resemble the following:

Lastly, configure the next project stage.

35. Drag and drop a **Set project stage** action onto the design canvas below the **Loop** action.
36. Configure the following setting:
    a.   In the **Stage** field, select **Execution**.

Next Steps

This workflow can now be associated to a new or existing Enterprise Project Type via Server Settings. See the Nintex Workflow for Project Server 2013 help file "Associating a Project Server Workflow with an Enterprise Project Type".

Note: Alternatively, the loop logic described in steps 9-34 may be used in an existing Project Server Workflow to ensure that project plans have been baselined.

Quick steps to run the workflow

1.   Create a new project using the Enterprise Project Type the workflow has been associated to.
2.   At the Scheduling stage, ensure that the project plan is created and DO NOT baseline the project plan.
3.   Submit the proposal in order to move it to the next stage. Notice that the project will not move to the next stage.
4.   Edit the project plan and set the project baseline.
5.   Submit the project in order to move it to the next stage. Notice that the project will now be able to move to the next stage.