NINTEX Workflow 2010 for Project Server

Tutorial:
# Sample Proposal Workflow

Content

Introduction

This tutorial will demonstrate how the 'Sample Proposal' workflow, provided by Microsoft Project Server 2010 and found in the Project Server SDK, can be recreated using Nintex Workflow for Project Server 2010.

 'Sample Proposal' directs a project through the complete project life cycle, from Create, Select, Plan, Manage, to Finish. All Stages, Project Properties and Project Detail Pages that are associated with the 'Sample Proposal' workflow are included by Microsoft Project Server 2010.
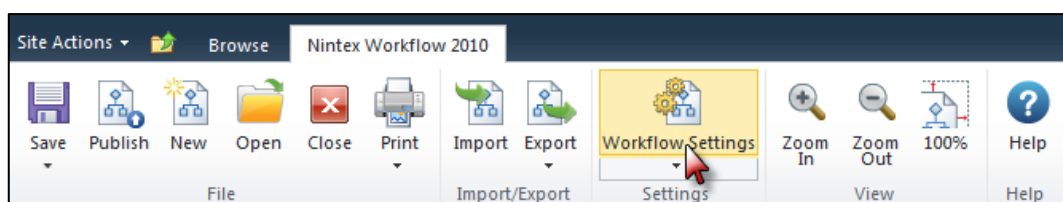
Getting Started

Open Nintex Workflow for Project Server 2010 workflow designer.

1. Navigate to the Project Web Application site.

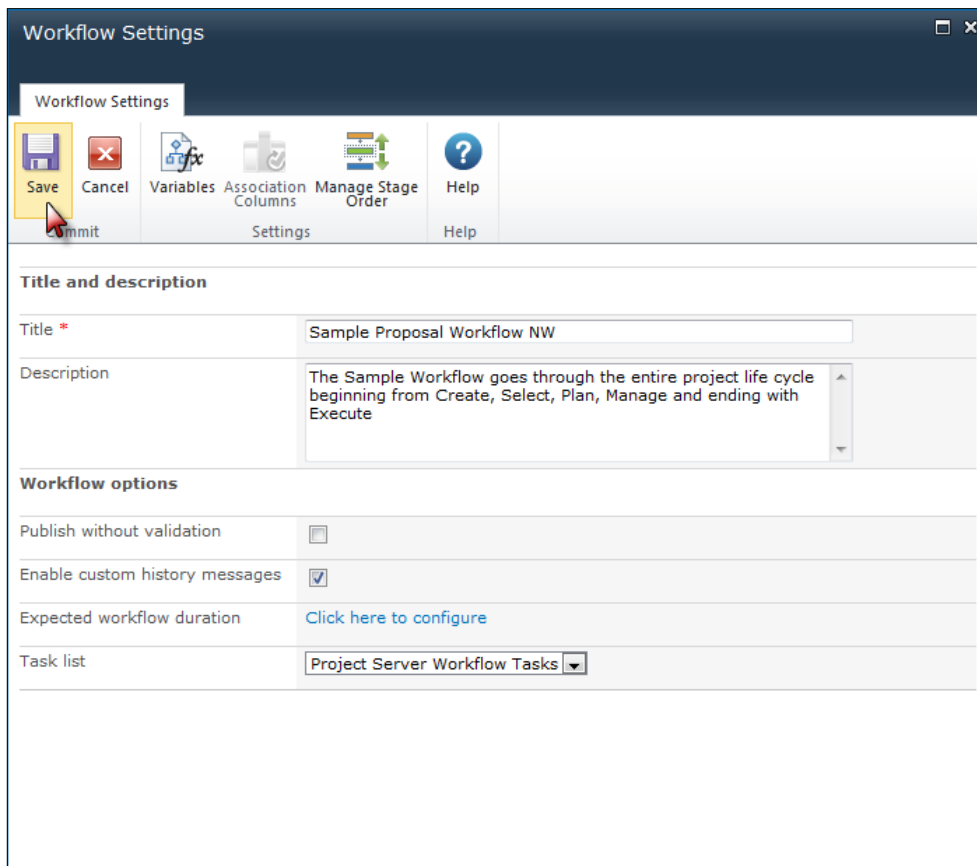2. Click Sites Actions > Project Server Workflow > Create Project Server Workflow.
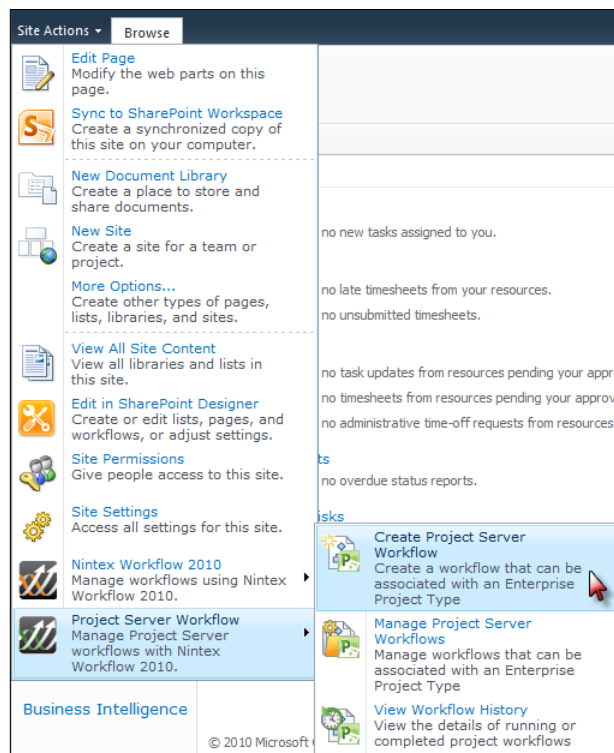
Configure the Workflow Settings and give the workflow a Title and Description.

3. In the Workflow Designer Ribbon, click on Workflow Settings.

4. In the Workflow Settings dialog, enter the Title and Description and click Save when completed.

Note: The task list called Project Server Workflow Tasks is the default task list used for Project server workflows that is accessible from the Workflow Approvals link in the PWA home page.

Designing the Sample Proposal Workflow

Phase One: Create

Start creating the workflow by dragging and dropping the workflow actions onto the design canvas.

1. Click and drag the Set project stage action from the toolbox and drop it onto the drop-zone ☐ on the design canvas.
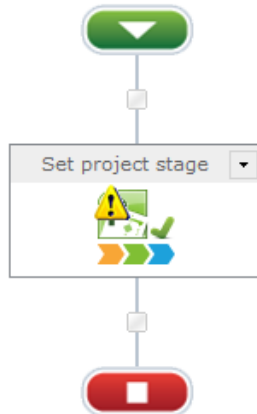
     a.   In the configuration settings, set the stage to Initial Proposal Details.

    Note: The first action in a Project Server workflow must be a Set project stage action.

This section of the workflow will query Project Server for the display and login names of the project owner (the user who created the proposal) and store them in variables that will be used throughout the workflow.
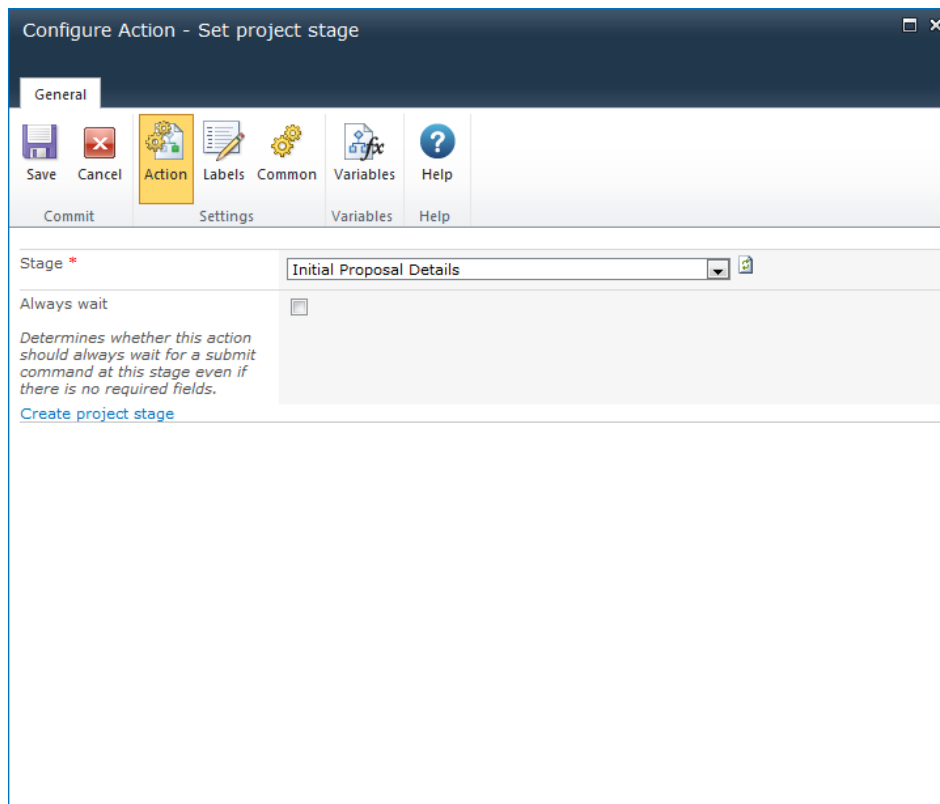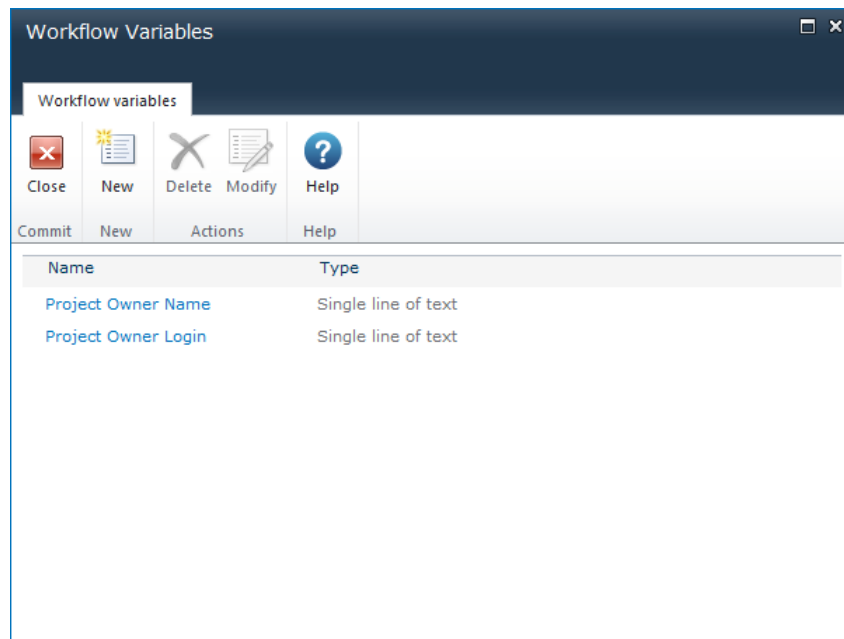
2. Create the following workflow variables to store the name and login of the project owner:
   a. Project Owner Name (Single line of text)
   b. Project Owner Login (Single line of text)

3. Add an Action set action to the designer canvas and rename the Action Title to **"Set Owner Information"**.
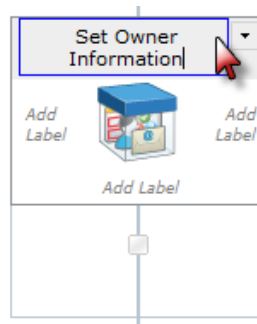
4. Drag and drop on the Read project property action within the Action set (Set Owner Information). In the Configuration settings, set the following:
   a. Project property: Project Owner Display Name
   b. Store result in: Project Owner Name

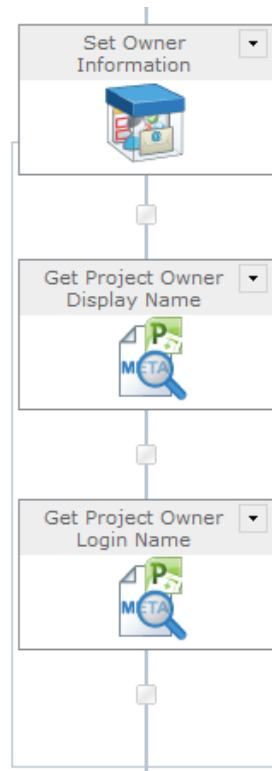5. Drag and drop on the Read project property action within the Action set (Set Owner Information). In the Configuration settings, set the following:
   a. Project property: Project Owner Login Name
   b. Store result in: Project Owner Login

The first condition in the business process is to automatically reject any proposal that has a cost of more than $1,000,000

6. Create the following workflow variable to be used for storing the comparison result:
   a. Is Too Expensive (Yes/No)

7. Drag and drop on the Compare project property action. In the Configuration settings, set the following:

   Note: The Compare project property action will return a True/False based on the comparison. The result can then be branched on in the workflow.

   a. Project property: Sample Proposal Cost
   b. Comparison Operator: Greater than
   c. Value: Value; 1000000
   d. Store result in: Is Too Expensive

8. Drag and drop the Set a condition action. In the Configuration settings, set the following:
   e. Condition: If any value equals value
   f. Where: Workflow Data; Is Too Expensive; equals; Value; Yes
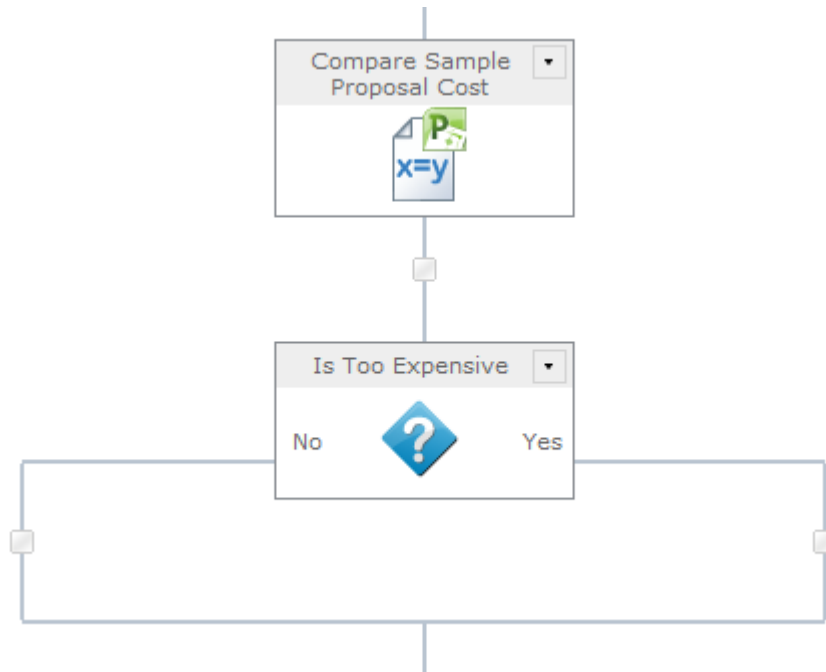

If the project cost is greater than $1,000,000, the workflow will send a notification to the Project Owner and end the workflow.

9. In the **"Yes"** branch, add and configure the following workflow actions:

   a. Drag and drop the Set project stage action. In the Configuration settings, set the following:
      i. Stage: Automatic Rejection

   b. Drag and drop the Set status information action. In the Configuration settings, set the following:
      i. Stage: Automatic Rejection
      ii. State: Leave current state

   c. Drag and drop the Send notification action. In the Configuration settings, set the following:
      i. To: Project Owner Login (workflow variable)
      ii. Subject: Proposal is too expensive
      iii. Body: Use inserted references to provide context to the email body.

Rich Text

Insert Reference

Dear Project Owner Name,

The proposal Context Item Display Name is too expensive.

View workflow status

View Item

Notes:

See the Nintex Workflow 2010 help file "Inserting reference fields" for more information about inserting references into a text field, including how to construct URLs.

In a Project Server workflow, using the 'Context Item Display Name' token will resolve to the name of the proposal, and the 'Context Item Url' will resolve to the URL of the proposal status page.

d. Drag and drop the End workflow action.
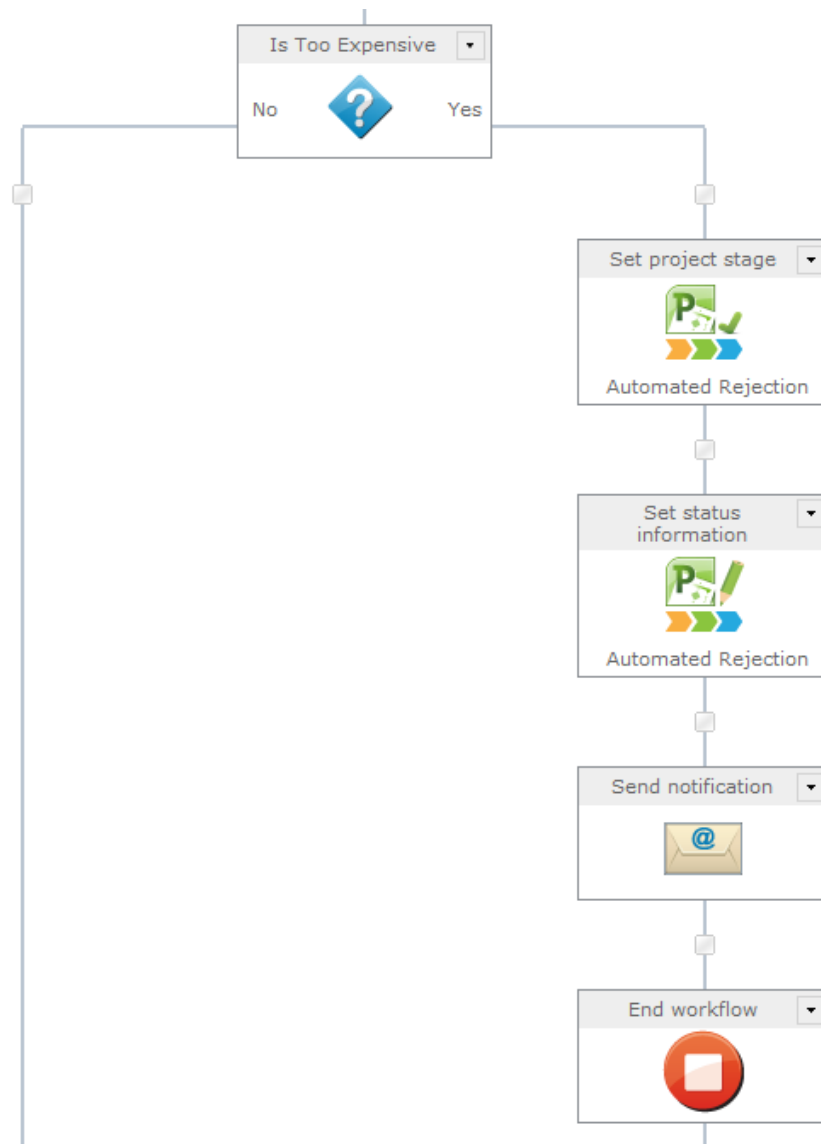
If the project cost is less than $1,000,000, the workflow will move to the 'Initial Review' stage for manual approval from the Team Leads (Project Server group).

10. In the **"No"** branch, add and configure the following workflow actions:

a. Drag and drop the Set project stage action. In the Configuration settings, set the following:
    i. Stage: Initial Review

b. Drag and drop the Set status information action. In the Configuration settings, set the following:
    i. Stage: Initial Review
    ii. State: Waiting for approval

    c. Drag and drop the Read project security group action. In the Configuration settings, set the following:
         i. Group name: Team Leads
        ii. Create a workflow variable Team Leads (single line of text) to store the users retrieved from the Project security group
   iii. Store result in: Team Leads
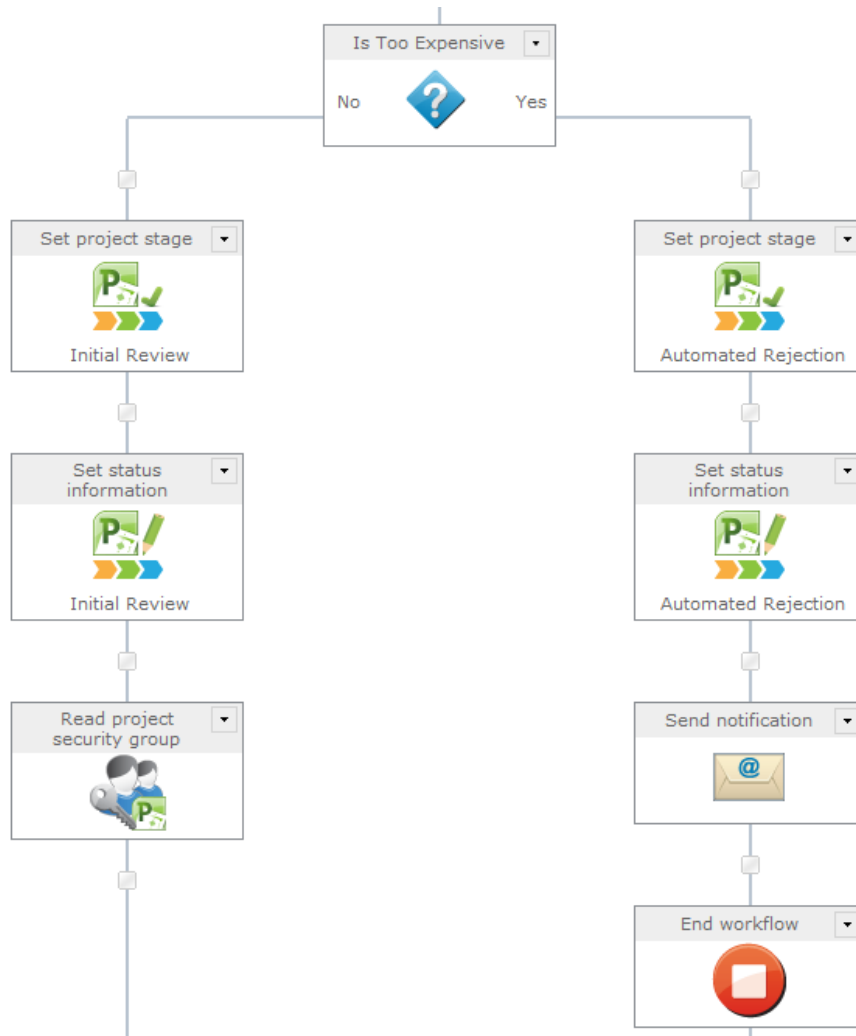
The workflow will request the proposal be approved by the users in the Team Lead group. This will be enclosed inside a 'Run if' action that is configured to only assign a task if the 'Read project security group' action determined that the Team Lead group did contain users, otherwise the section will be bypassed.

    d. Drag and drop the Run if action. In the Configuration settings, set the following:
         i. Condition: If any value equals value
        ii. Where: Workflow Data; Team Leads; is not empty

An 'Assign a flexi task' action is used to request approval from a user. It will send the user a notification requesting their response with a URL to a task page for them to submit their decision. The action can be configured to change the rules of the approval process (i.e. all must respond, all must agree, etc.)
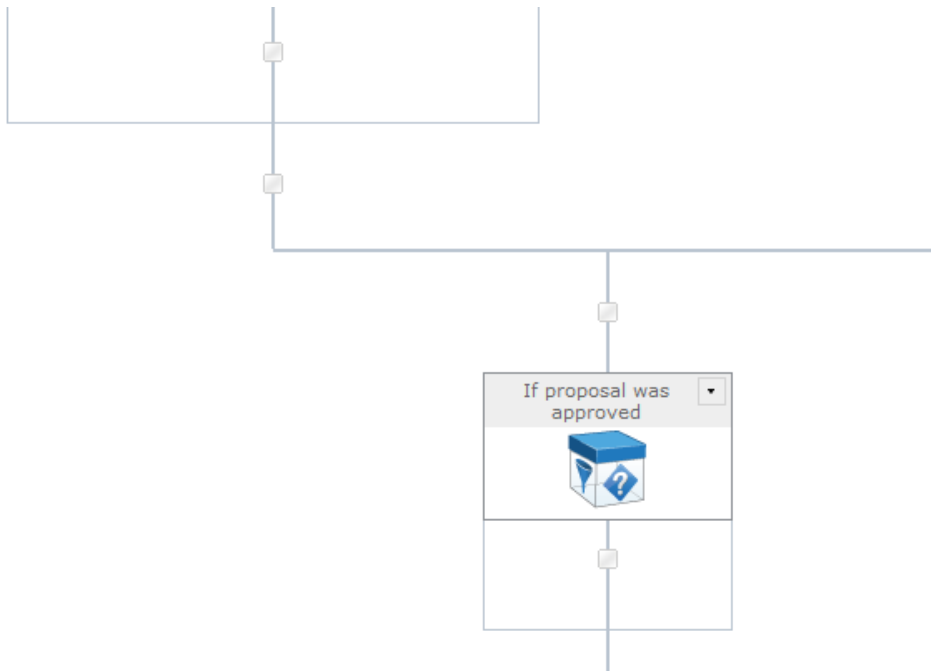
e. Inside the Run if action, drag and drop the Assign Flexi task action, configure the options as follows:
   i. Assignees: Team Leads
   ii. Leave all other options as default

f. Create a workflow variable Approved (Yes/No) to store the outcome of the Reject/Approve branches of the Assign Flexi task.

g. In the Reject branch, drag and drop the Set a variable action. In the Configuration settings, set the following:
   i. Set: Approved; Value; No

h. In the Approve branch, drag and drop the Set a variable action. In the Configuration settings, set the following:
   i. Set: Approved; Value; Yes
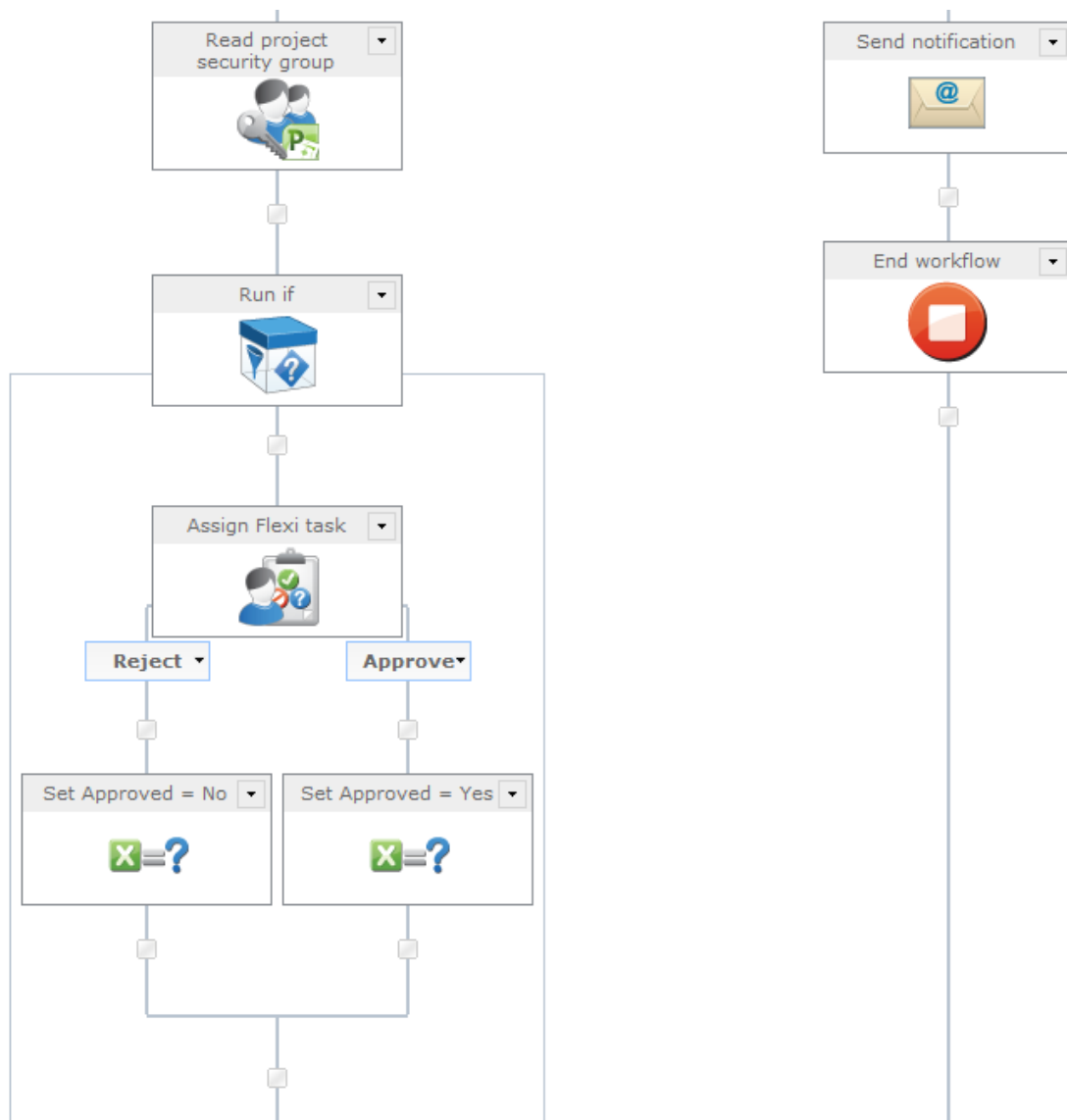
If the 'Initial Review' stage is approved, the workflow will move to the 'Proposal Details' stage to request for more information.

11. Drag and drop a Run if action after the Set a condition and all its branching actions. Rename the Action Title to **"If proposal was approved"**. In the Configuration settings, set the following:
    a. Condition: If any value equals value
   b. Where: Workflow Data; Approved; equals; Value; Yes

12. Inside the Run if action, drag and drop the Set project stage action. In the Configuration settings, set the following:
    a. Stage: Proposal Details

Another approval process will now be required to review the additional project details.

13. After the Set project stage, drag and drop another Set project stage action. In the Configuration settings, set the following:
    a. Stage: Selection Review

14. Drag and drop the Set status information action. In the Configuration settings, set the following:
    a. Stage: Selection Review
    b. State: Waiting for approval
    c. Status information: Project is waiting for approval
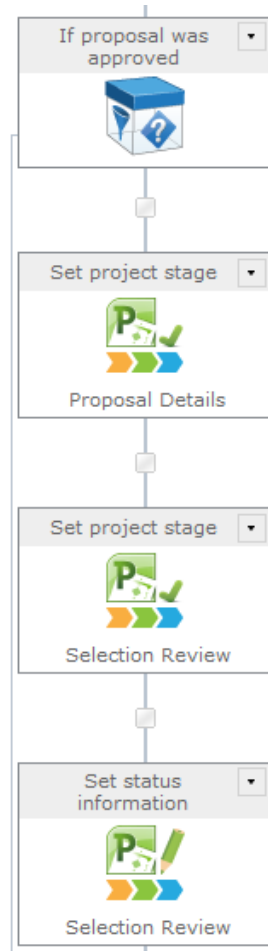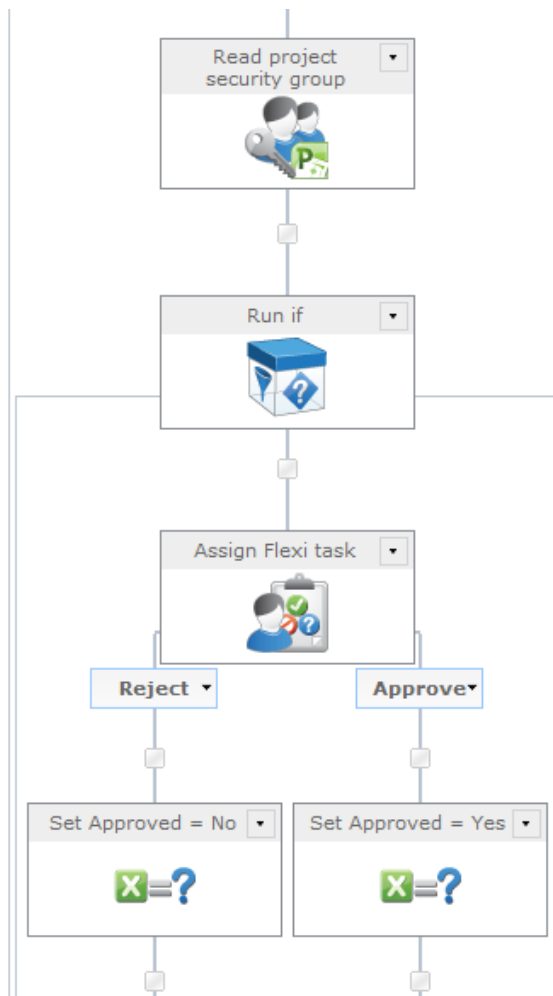

15. Repeat the approval process from step 9c. Copy the Read project security group action. This will ensure the latest list of team leads is always retrieved before an approval process.
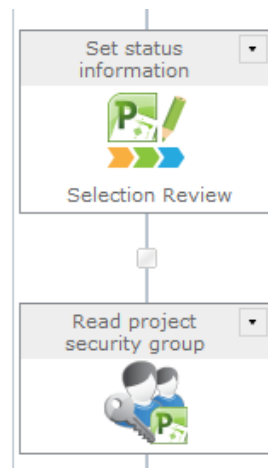
16. Next copy the Run if action from Step 9d (which includes 9e-9h). Note: If the approval notification was configured in step 9e, remember to make any necessary changes.


Thus far the approval of the Initial Review and Selection Review stage has been configured. Next, if the Initial Review or Selection Review is rejected (Approved variable is 'No'), the proposal will be rejected and the workflow ended.

17. After the Run If (If proposal was approved), drag and drop a Run if action and rename the Action title to **"If proposal was not approved"**. In the Configuration settings, set the following:
    a. Condition: If any value equals value
    b. Where: Workflow Data; Approved; equals; Value; No
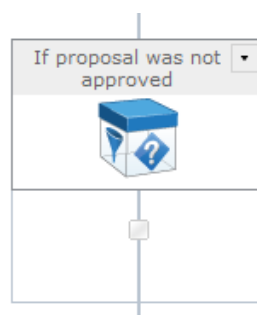
18. Inside the Run if action, drag and drop the Set project stage action. In the Configuration settings, set the following:
    a. Stage: Rejected

19. Drag and drop the Send notification action. In the Configuration settings, set the following:
    a. To: Project Owner Login (workflow variable)
    b. Subject: Proposal was rejected
    c. Body: Create a message using the Project Owner Name , Context Item Display Name (name of the proposal and Context Item URL (URL to the status)

20. Drag and drop the End workflow action.
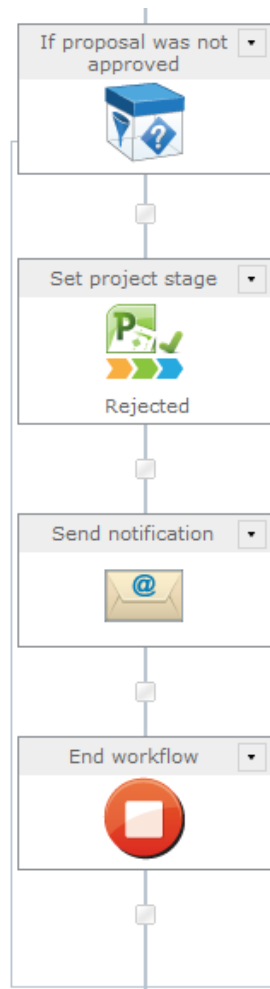
## Phase Two: Select

If the Selection Review stage is approved (Approved variable is 'Yes'), the proposal will move to the next phase of the workflow where proposals are selected to be moved forward to the next stage.

1. After the Run if (If proposal was not approved) action, drag and drop the Set project stage action, configure the options as follows:
    a. Stage: Proposal Selection

At this point the workflow will wait for the proposal to undergo analysis to determine if it has been committed to or not.

2. Drag and drop the Wait for commit action.
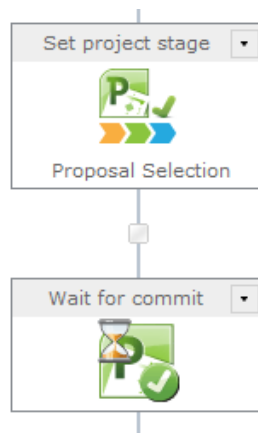
Once the proposal has been committed, the workflow continues looping until it has been selected in the Portfolio Analyses tool. The proposal will wait in this loop indefinitely until it has been selected in both the Analyze Cost screen and in the Analyze Resources screen. The workflow logic takes into account that the proposal could be automatically in or forced in.

The project properties representing the selection result contain numeric values representing 'in', 'forced in', 'out' and 'forced out'.

By default, Nintex Workflow for Project Server 2010 has the following workflow constants created and set:
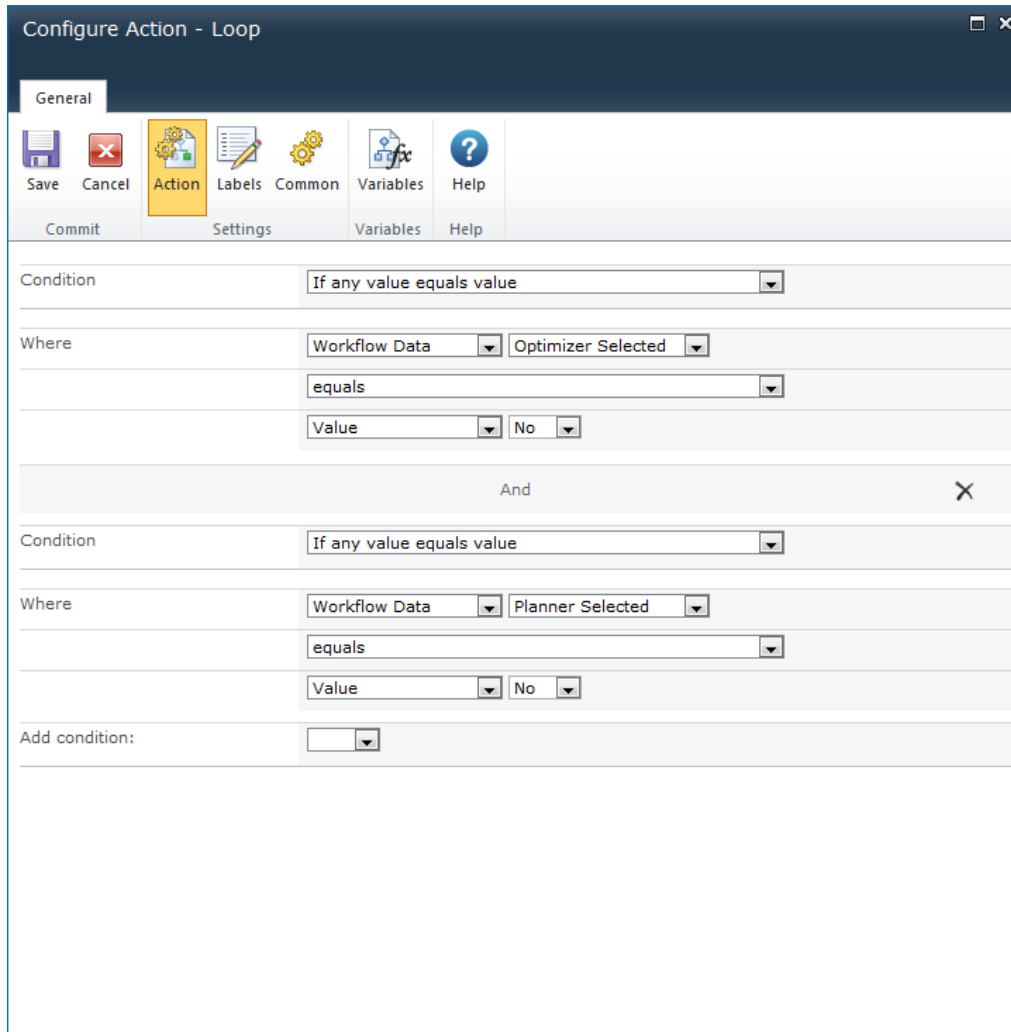
    a. Decision Result Value – Forced In (Number): 0
    b. Decision Result Value – Forced Out (Number): 1
    c. Decision Result Value – In (Number): 3
    d. Decision Result Value – Out (Number): 2

At this point the workflow will need to wait until the proposal is selected.

3. Create the following workflow variables:
    a. Optimizer Selected (Yes/No)
    b. Planner Selected (Yes/No)

4. Drag and drop the Loop action and rename the Action Title to **"While Opti**mizer and **Planner not selected"**. Add the following workflow actions and configure them as follows:
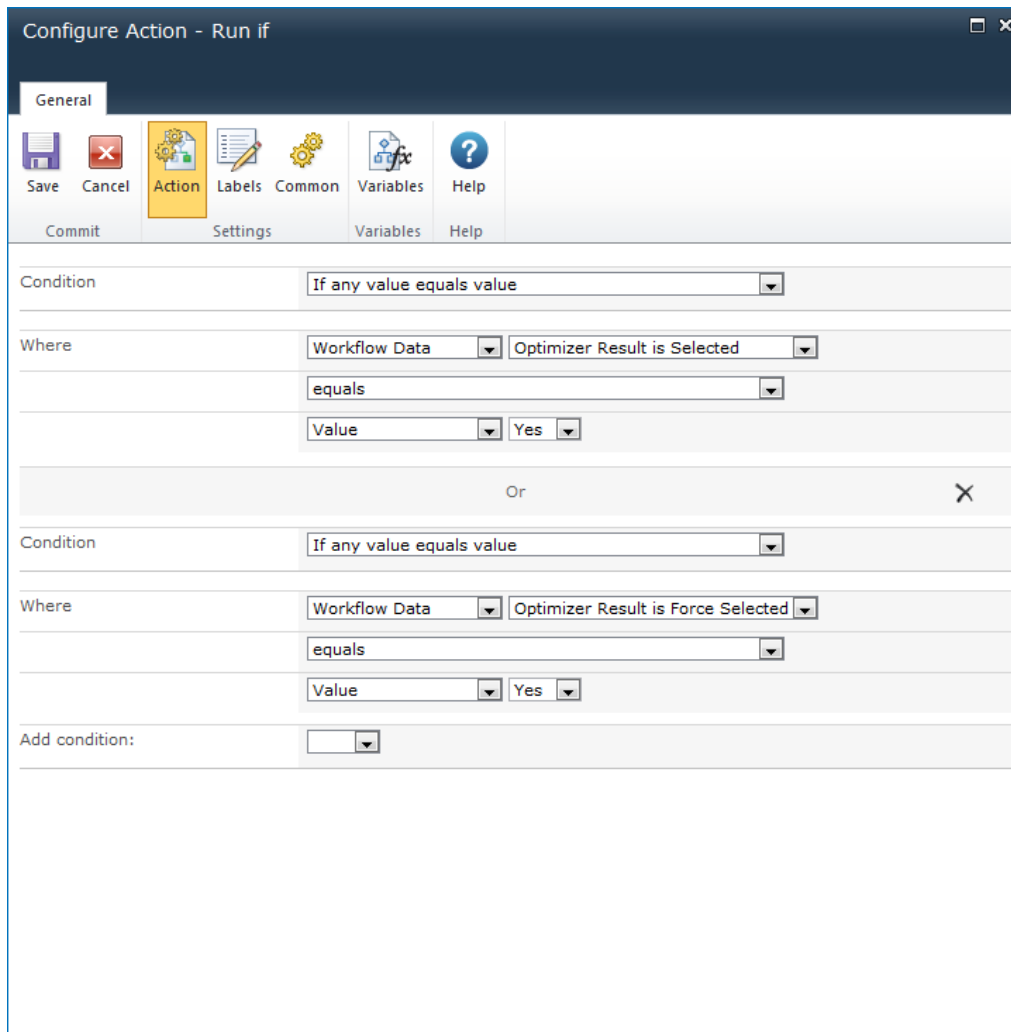
The Optimizer Result project property will be compared with the value for 'in' and 'forced in' to see if the proposal has been selected by either means

a. Create the following workflow variables:
    i. Optimizer Result is Selected (Yes/No)
    ii. Optimizer Result is Force Selected (Yes/No)

b. Drag and drop the Compare project property action and rename the Action Title to **"Optimizer Result is Selected?"**. In the Configuration settings, set the following:
    i. Project property: Optimizer Decision
    ii. Comparison operator: Equals
    iii. Value: Workflow Constants; Decision result –in
    iv. Store result in: Optimizer Result Is Selected

c. Drag and drop the Compare project property action and rename the Action Title to **"Optimizer Result is Force Selected?"**. In the Configuration settings, set the following:
    i. Project property: Optimizer Decision
    ii. Comparison operator: Equals
    iii. Value: Workflow Constants; Decision result –forced in
    iv. Store result in: Optimizer Result Is Force Selected
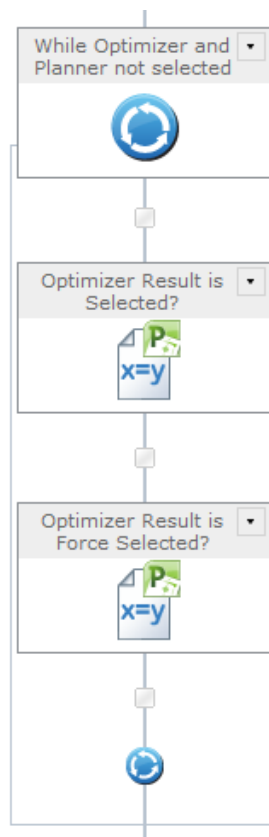
d. Drag and drop the Run if action and rename the Action Title to **"Check optimizer selected results"**. In the Configuration settings, set the following:
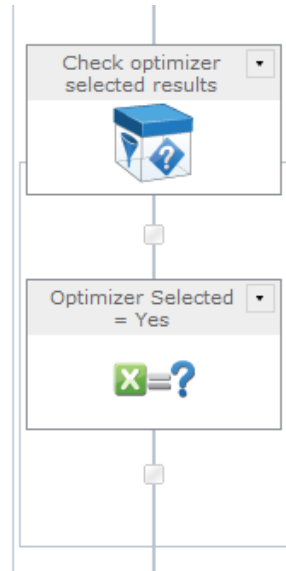
e. Inside the Run if (Check optimizer selected results) action, drag and drop on the Set variable action and rename the Action Title to **"Optimizer selected is Yes"**. In the Configuration settings, set the following:

  i. Set "Optimizer Selected" to "Yes"

If the proposal was not selected, it will move to the 'Not Selected' stage and wait for the next commit.

f. After the Run if (Check optimizer selected results) action, drag and drop the Set a condition action and rename the Action Title to **"If optimizer selected"**. In the Configuration settings, set the following:
    i. Condition: If any value equals value
    ii. Where: Workflow Data; Optimizer Selected; equals; Value; Yes

g. In the "No" branch, add and configure the following workflow actions:

    i. Drag and drop the Send notification action. In the Configuration settings, set the following:
        1. To: Project Owner Login (workflow variable)
        2. Subject: Your proposal was not selected
        3. Body: Create a message using the Project Owner Name , Context Item Display Name (name of the proposal and Context Item URL (URL to the status)

    ii. Drag and drop on the Set project stage action. In the Configuration settings, set the following:
        1. Stage: Not selected

    iii. Drag and drop a Wait for commit action.
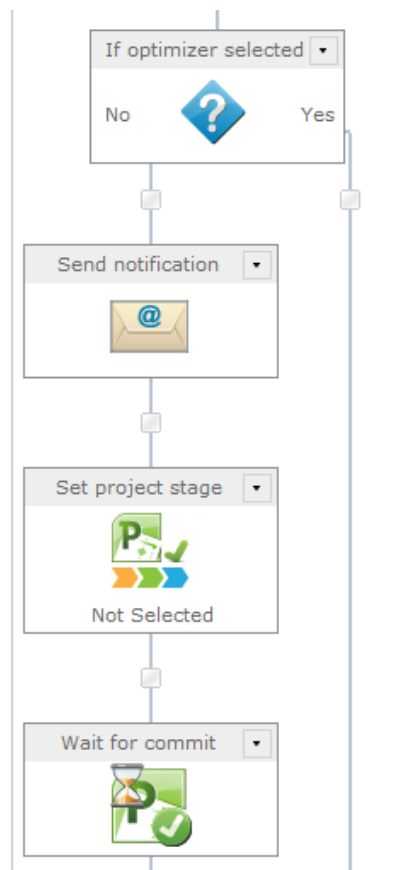
Phase Three: Plan

If the proposal was selected, the proposal will enter the "resource planning" stage.

1. In the Set a condition (If optimizer selected) action "Yes" branch, add and configure the following workflow actions:

   a. Drag and drop the Send notification action. In the Configuration settings, set the following:
      i. To: Project Owner Login (workflow variable)
      ii. Subject: Proposal is in Planning Stage
      iii. Body: Create a message using the Project Owner Name , Context Item Display Name (name of the proposal and Context Item URL (URL to the status)

   b. Drag and drop the Set project stage action. In the Configuration settings, set the following:
      i. Stage: Resource planning

   c. Drag and drop the Wait for commit action.
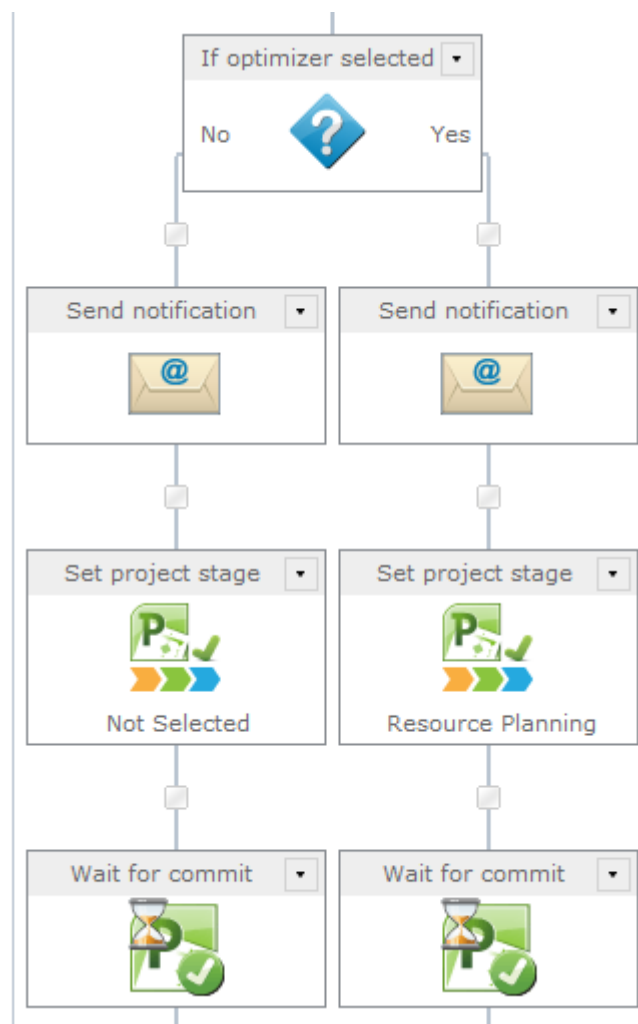
The Planner Result project property will now be compared to the value for 'selected' and 'forced selected' to see if the proposal was selected at the 'Analyze Resources' screen.
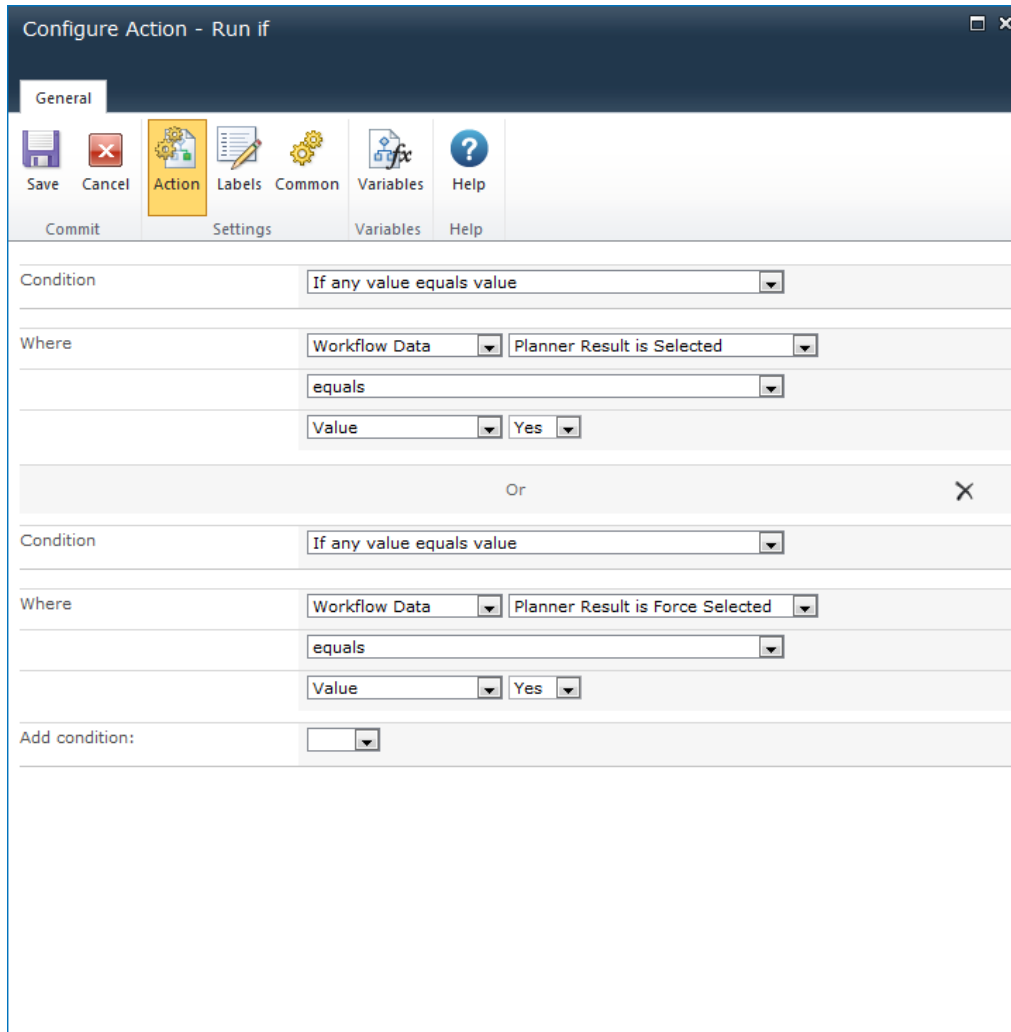
d. Create the following workflow variable:
    i.   Planner Result is Selected (Yes/No)
    ii. Planner Result is Force Selected (Yes/No)

e. After the Wait for commit action in the "Yes" (Resource Planning) branch, drag and drop a Compare project property and rename the Action Title to **"Planner Result is Selected?"**. In the Configuration settings, set the following:
    i.   Project property: Planner Decision
    ii.   Comparison operator: Equals
    iii. Value: Workflow Constants; Decision result –in
    iv.   Store result in: Planner Result is Selected

f. Drag and drop the Compare project property, rename the Action Title to **"Planner Result is Force Selected?"**. In the Configuration settings, set the following:
    i.   Project property: Planner Decision
    ii.   Comparison operator: Equals
    iii. Value: Workflow Constants; Decision result –forced in
    iv.   Store result in: Planner Result is Force Selected
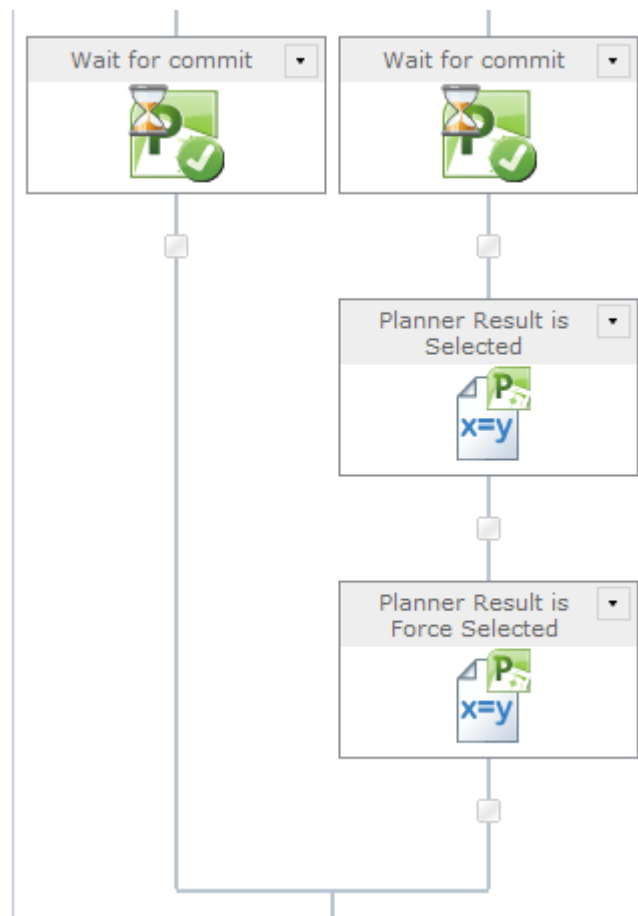
g. Drag and drop a Run if action and rename the Action Title to **"Check planner selected results"**. In the Configuration settings, set the following:
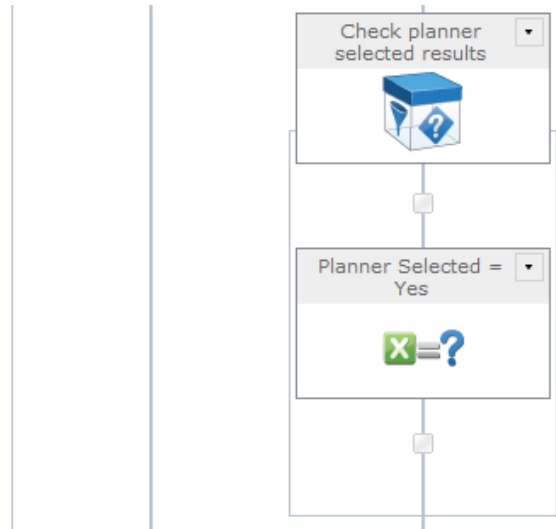
h. Inside the Run if (Check planner selected results) action, drag and drop on a Set variable action and rename the Action Title to **"Planner selected is Yes"**. In the Configuration settings, set the following:

   i. Set "Planner Selected" to "Yes"

Once the proposal is selected and planned, the workflow will move out of the loop onto the Scheduling stage.

2. After the Loop (While Planner and Optimizer not selected), drag and drop on a Set project stage. In the Configuration settings, set the following:

   a. Stage: Scheduling
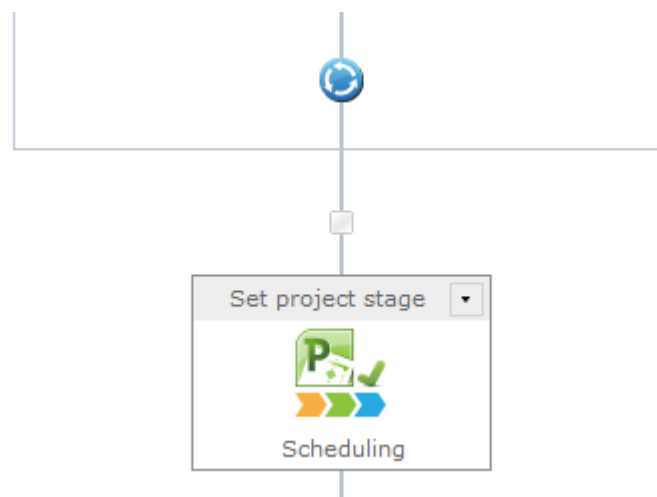
3. Repeat the approval process from Phase One: Create, step 9. Copy the Read project security group action. This will ensure the latest list of team leads is always retrieved before an approval process.

4. Next copy the Run if action from Phase One: Create, Step 9d (which includes 9e-9h). Note: If the approval notification was configured in step 9e, remember to make any necessary changes.
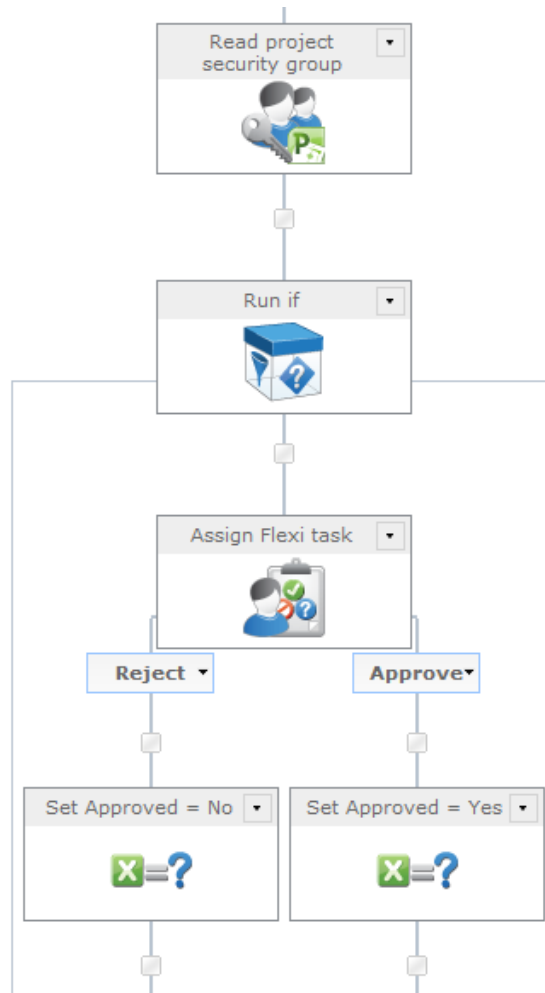
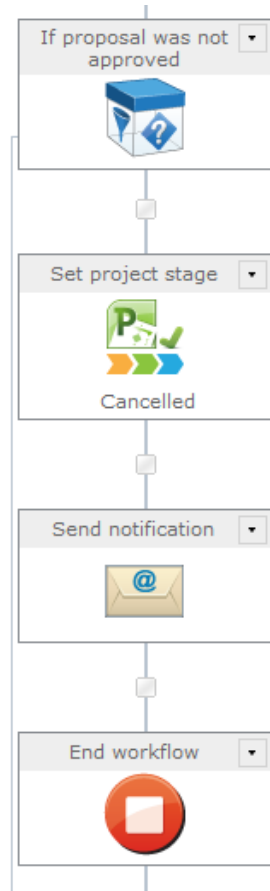If the approval fails, the proposal will be cancelled.

5. Drag and drop a Run if action and rename the Action Title to **"**If proposal was **not approved"**. In the Configuration settings, set the following:
   a. Condition: If any value equals value
   b. Where: Workflow Data; Approved; equals; Value; No
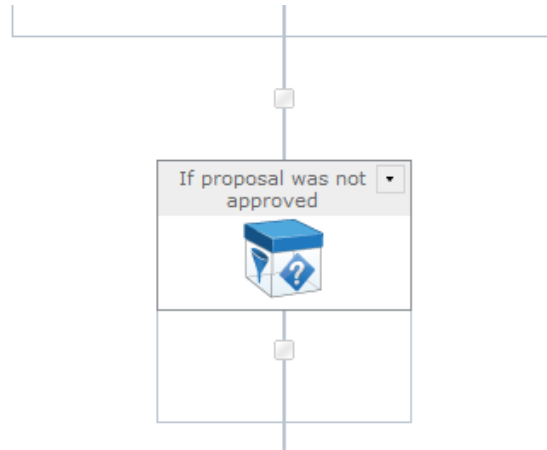
6. Inside the Run if (If proposal was not approved) action, drag and drop the Set project stage action. In the Configuration settings, set the following:
   a. Stage: Cancelled

7. Drag and drop the Send notification action. In the Configuration settings, set the following:
   a. To: Project Owner Login (workflow variable)
   b. Subject: Proposal was cancelled
   c. Body: Create a message using the Project Owner Name , Context Item Display Name (name of the proposal and Context Item URL (URL to the status)
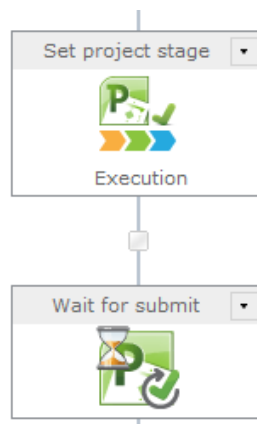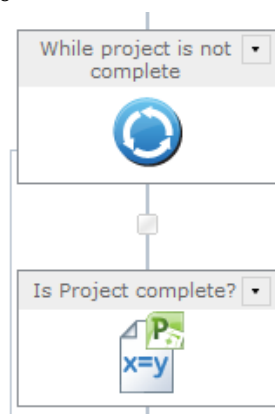
8. Drag and drop the End workflow action.

Phase Four: Manage

1. After the Run if (If proposal was not approved), drag and drop the Set project stage action. In the Configuration settings, set the following:
   a. Stage: Execution

2. Drag and drop the Wait for submit action.

The workflow will check each time the project is submitted to see if the completed percentage is 100%.

3. Create the following workflow variable:
   a. Is Project Complete (Yes/No)

4. Drag and drop the Loop action and rename the Action Title to **"While project is not complete"**. In the Configuration settings, set the following:
   a. Condition: If any value equals value
   b. Where: Workflow Data; Is Project Complete; equals; Value; No

5. Inside the Loop (While project is not complete) action, drag and drop on the Compare project property action and rename the Action Title to **"Is Project Complete?"**. In the Configuration settings, set the following:
   a. Project property: Percentage Complete
   b. Comparison operator: Equals
   c. Value: Value; 100
   d. Store result in: Is Project Complete

6. After the Compare project property (Is Project complete?) action, drag and drop a Run if action and rename the Action Title to **"If project is not complete"**. In the Configuration settings, set the following:
    a. Condition: If any value equals value
    b. Where: Workflow Data; Is Project Complete; equals; Value; No

7. Inside the Run if (If project is not complete) action, drag and drop on the Wait for submit action.
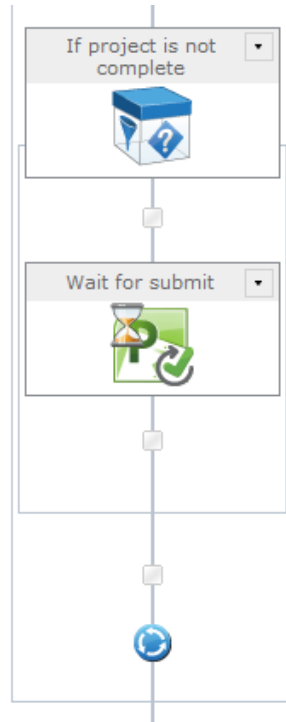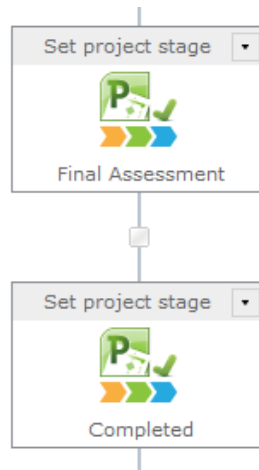
Phase Five: Finished

Once the project status is 100%, the final two stages will be processed.

1. Drag and drop the Set project stage action. In the Configuration settings, set the following:
   a. Stage: Final Assessment

2. Drag and drop the Set project stage action. In the Configuration settings, set the following:
   a. Stage: Completed

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 2010 help file "Associating a Project Server Workflow with an Enterprise Project Type".