



Guide to Automating Workflows Quickly and Easily

Part 3

**work
flow
for
every
one**

NINTEX

Table of Contents

Introduction	3
Handling the Full Range of Project, Portfolio, and Collaboration Workflows	3
Demand Management Workflows.....	3
Event Driven Workflows	3
Collaboration Workflows	4
Project Workspace Collaboration	5
Simple Document Approval	7
Synchronise Documents with Office 365	15
Scheduling the Workflow.....	23
Extending the Example	23
Conclusion.....	24

Introduction

Welcome to our Guide to *Automating Workflows Quickly and Easily: Nintex Workflow for Project Server 2010*. We will look at Nintex Workflow for Project Server 2010 and the ways it can empower your Project Managers to automate project, portfolio and collaboration workflows quickly and easily.

This guide is the third in a three-part series discussing each of the workflow types provided by Nintex Workflow for Project Server 2010 providing easy to follow step-by-step examples.

Nintex Workflow for Project Server makes the lives of Project and Portfolio Managers a lot easier, as they use a simple web-based editor to build and maintain workflows, removing the reliance on complex and costly development cycles. Nintex Workflow for Project Server is a graphical, web-based workflow designer built for the SharePoint and Project Server platforms. The key capabilities outlined in this guide are:

Project, Portfolio, and Collaboration Workflows	Demand Management Workflows Event Driven Workflows Project Workspace Workflows
Quick and Easy Functionality	Browser Based Designer No Code Required Drag and Drop
Connectivity with Other Systems	Query Project Data Connect to Internal and External Systems Integrate with Office 365

Figure 1: The pillars of Nintex Workflow for Project Server

Handling the Full Range of Project, Portfolio, and Collaboration Workflows

Project and Portfolio Managers have a range of different needs for automating tasks and processes. A workflow tool should fulfill the needs for designing processes related to demand management, event driven, and collaborative workflows.

Demand Management Workflows

Demand Management Workflows allow the orchestration of an organization’s complete governance process using Project Servers demand management capabilities. Through these workflows, project ideas and projects can be guided through a governance process, whilst applying business rules, validations and approvals along the way.

Event Driven Workflows

Nintex Workflow for Project Server introduced the ability to associate workflows with Project Servers event handling infrastructure. Project Server has the ability to associate bespoke code or assemblies against events that are raised by Project Server, such as Project Published, Project Saved or Custom Field checked in etc. With Nintex Workflow for Project Server 2010, it is now possible to associate a Nintex Workflow against a subset of events within the event handler infrastructure

quickly and easily. Nintex Workflow for Project Server Event Driven Workflows allows you to create workflows that automate scenarios such as:

- Alert Project Managers when a resource is added or activated
- Notify management when a project is published
- Query line-of-business data when a project starts
- Sync custom fields and lookup tables across Project Web App sites
- Keep team members in the loop through integration with SharePoint and much more.

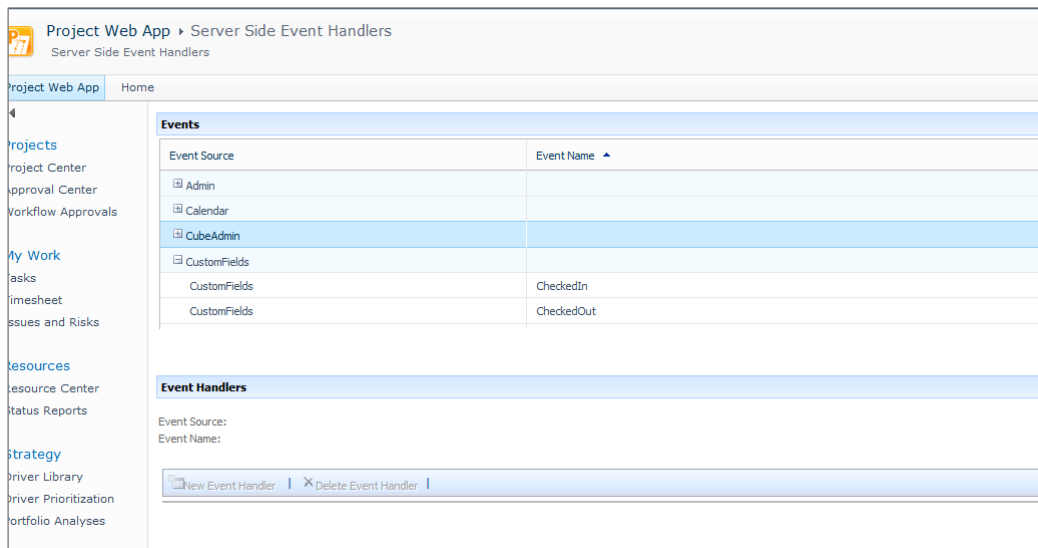


Figure 2: Project Server Service Side Event Handlers

Collaboration Workflows

Nintex Workflow for Project Server 2010 is built on top of Nintex Workflow 2010, and provides a number of capabilities that can be leveraged within the project workspaces; ranging from document approval workflows and notifications, to stakeholders when certain risks or issues are logged, through to synchronizing content from the workspace to an external Office 365 account for collaboration with external parties.

In addition, Nintex Workflow for Project Server also leverages a number of core capabilities from Nintex Workflow 2010, including:

- Leverage pre-built workflow actions ranging from provisioning site collections, list actions, interactions with external systems such as Exchange, Lync and SQL Server directly from your workflow .
- The ability to build collaboration workflows for document approvals and other line of business applications within your SharePoint farm.
- Powerful web based drag- and- drop editing, allowing workflows to be developed in minimal time and effort.
- Powerful debugging capabilities.
- Integration with other Nintex products such as Nintex Live and Nintex Forms.

Project Workspace Collaboration

Project Server 2010 is intrinsically linked with SharePoint 2010, from the Project Web Application being hosted within SharePoint 2010, through to the various linked project workspaces that a user can choose to create for each project for collaboration, storage of project artefacts such as document, risks and issues.

As these project workspaces evolve during the lifetime of the project, it can become necessary to implement controls directly within the workspace, much like the governance workflows that have been implemented within the PWA instance itself. For example, before a document is set to approved, it may be necessary for the document to be reviewed by the project team members and the outcome of those reviews to be recorded for audit purposes. Similarly, some organizations may wish for issues or risks to be reviewed by the Project Manager prior to the item being accepted onto the register and managed.

Typically, the type of requirements for workflow can vary greatly between different projects; therefore it is important to provide the project team with a tool that enables them to quickly and easily build workflow solutions with a minimum of fuss and involvement from an administrator.

Being built on top of Nintex Workflow 2010, Nintex Workflow for Project Server 2010 exposes all the capabilities of Nintex Workflow 2010¹ within the Project Workspaces, along with the following Nintex Workflow for Project Server 2010 actions:

- **Query Project Server** – Allows the workflow to query the Project Server PSI Read methods
- **Update Project Properties** – Allow project level properties to be updated

These actions allow Project Workspace workflows to be integrated directly into Demand Management workflows allowing an unprecedented level of integration. For example, with these capabilities it is now possible to 'pause' a Demand Management workflow until a document within the project workspace has been approved.

With the increased uptake of cloud based services, Nintex Workflow for Project Server 2010 is ideally placed to leverage these capabilities, specifically in Project Workspace scenarios. With Nintex Live, an add in for Nintex Workflow 2010, it is possible to incorporate workflow actions that enable integration with cloud based services, ranging from Bing, Office 365 and SkyDrive amongst others into your workflow, with new cloud based actions being added regularly.

¹ The exact capabilities of Nintex Workflow 2010 available will depend on the edition you have licenced. For full details of the different versions, please refer to www.nintex.com.

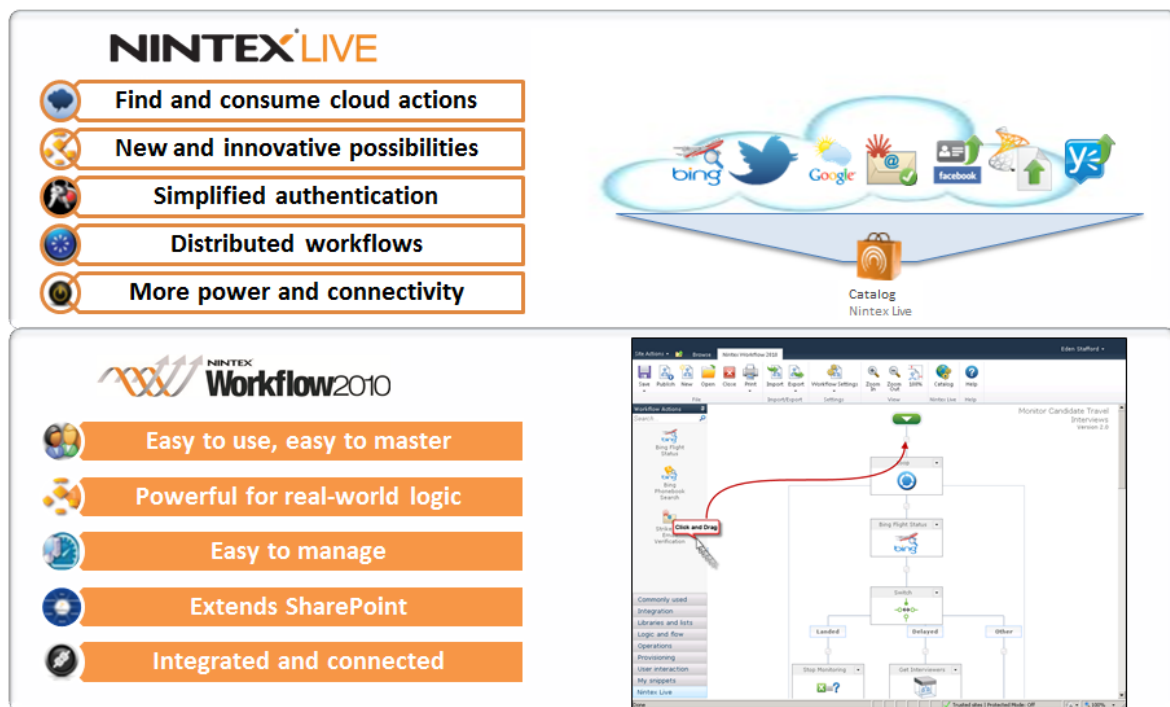


Figure 3: Nintex Live

To demonstrate the power and flexibility of Nintex Workflow 2010 within a Project Workspace, we are going to create two example workflows:

- **Simple Document Approval** – An approval workflow that will determine and route the approval to the project manager as defined in Project Server.
- **Synchronizing Project Documents to Office 365** – A workflow that will synchronize selected documents within a project workspace to an external Office 365 SharePoint site for distribution to external parties.

Simple Document Approval

This workflow will be initiated by a team member when they wish to have a document approved by the Project Manager. The workflow will need to determine the Project Manager as indicated in Project Server and then route an approval with a simple approve/reject outcome.

Before starting, you will need to activate the Nintex Workflow 2010 features in your Project Workspace template. To complete this example, either manually enable the feature within Site Features, or create a new Project Workspace template with the feature enabled and use this to create your workspace site for the example.

1. Navigate to the document library that you wish to enable the workflow on. In this case, the Project Documents Library.

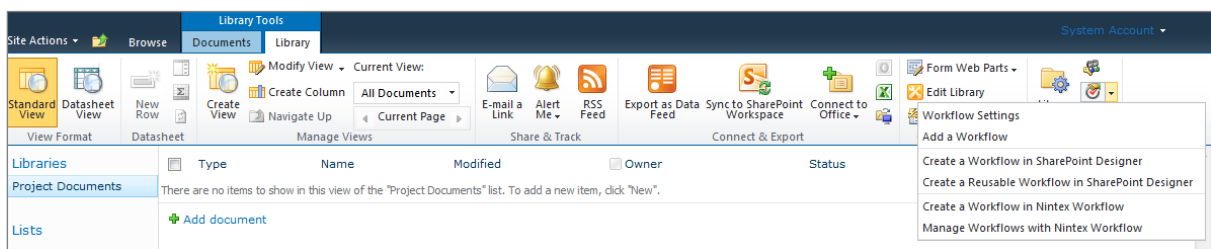


Figure 4: Create a Workflow in Nintex Workflow

2. Select the Library Ribbon, click on **Workflow Settings** and then **Create a Workflow in Nintex Workflow**. This will open the Nintex Workflow Designer.

To initiate the workflow, we will configure the workflow to add a menu item to the context menu in the workspace.

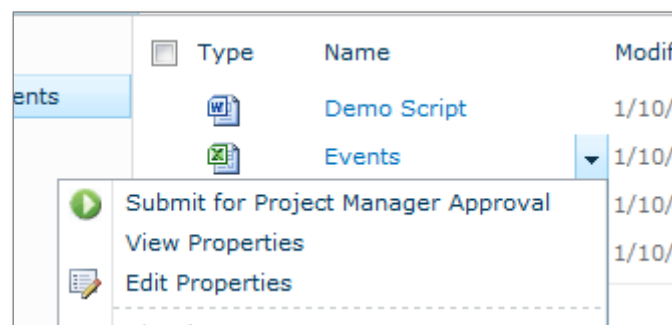


Figure 5: Submit for Project Manager Approval

3. To configure the context menu, select **Workflow Settings** and check the **Enable workflow to start from the item menu** option. Provide the **Menu item label** and a different image if you wish.

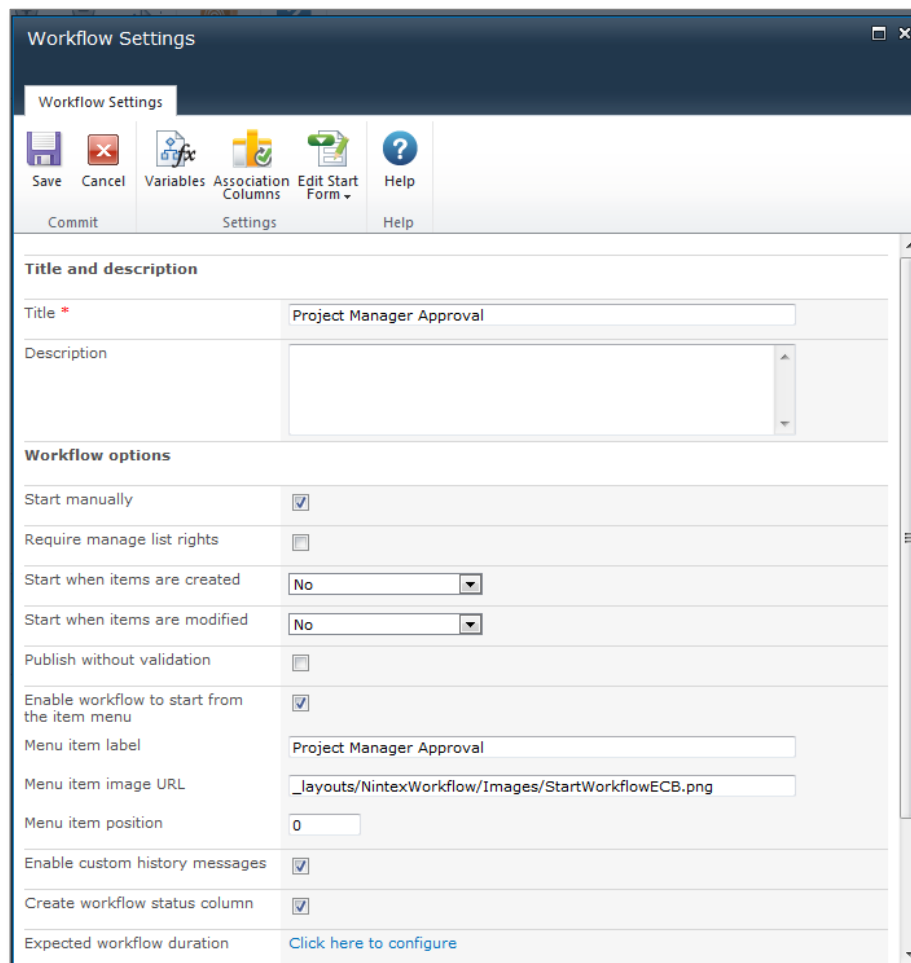


Figure 6: Workflow Settings

4. For this example it is necessary to leverage the **Execute SQL** action instead of a **Query project server** action. The reason for this is as the workflow is running in the workspace; there is no simple way to determine the **ProjectUID** the workspace is associated with via the Project Server PSI. But, due to Project Server’s Reporting Database, it is possible to run a direct SQL query against the database to retrieve the project which has the same URL of the workspace the workflow has been started in. In this case we will use the Reporting database to run a query to retrieve the Resource Name for the project with the same URL as the workspace (as stored in the Nintex Common variable **WebUrl**).

Configure the Execute SQL action as follows:

- a. **Connection String** should equal the connection string to connect to your Project Server reporting database, in this example our connection string is, however yours may have a different data source and catalog
Data Source=demo2010a; Initial Catalog=ProjectServer_Reporting; Integrated Security=SSPI
- b. Ensure **Windows Authentication** is checked

c. Query:

```
select epmr.ResourceName from msp_epmproject epmp
inner join msp_epmresource epmr on epmp.ProjectOwnerResourceUID =
epmr.ResourceUID where
epmp.ProjectWorkspaceInternalHref='{Common:WebUrl}'
```

d. In the **Store result in** menu select a new single line of text variable called **ProjectOwnerName**

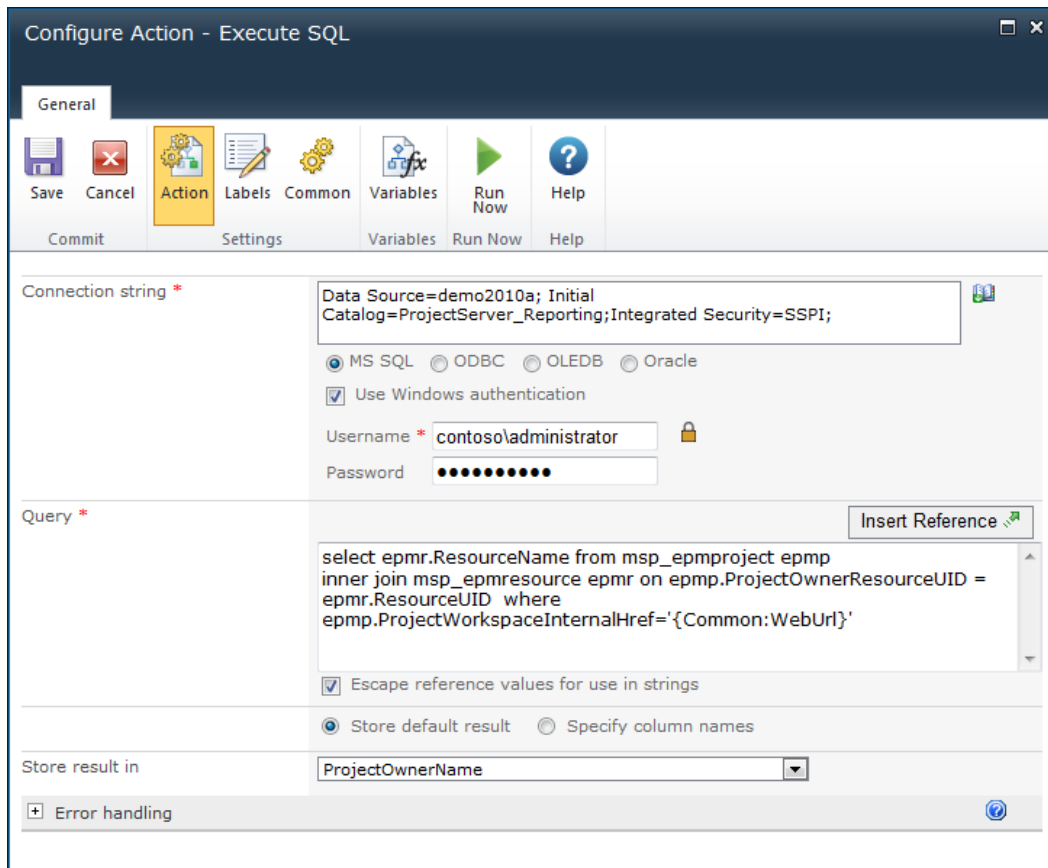


Figure 7: Configure Execute SQL Action

5. Once the Project Owner is known, all that is required is to request approval from the owner using a **Request Approval** action.

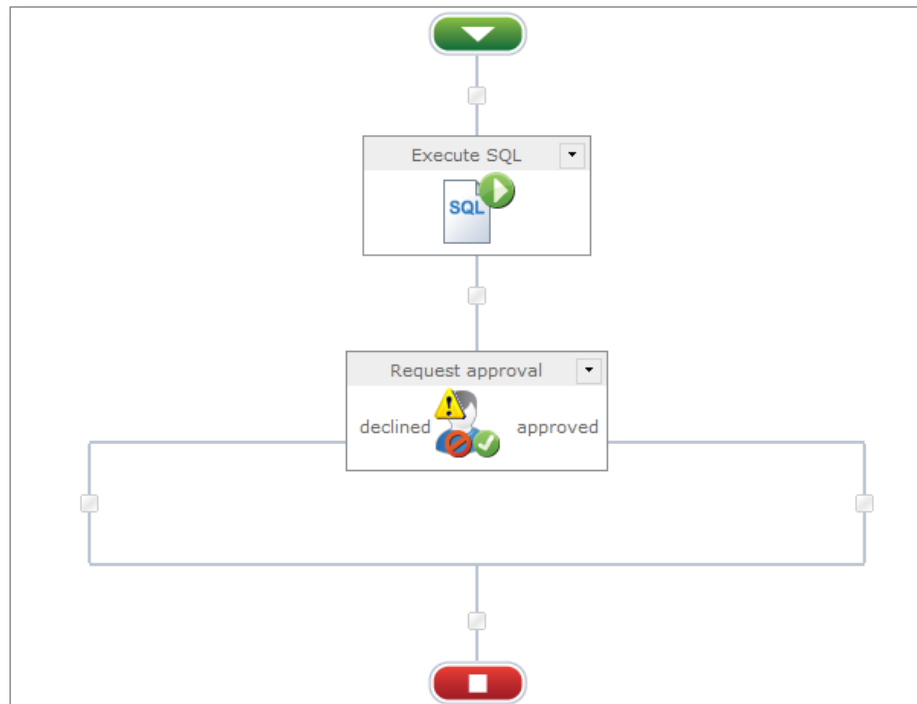


Figure 8: Add Request Approval Action

This action has two outcome branches, one for approved and another for declined and will look like the above image once added to the design surface.

- Double-click on the Request Approval action and configure it as follows, making sure the approver uses the **ProjectOwnerName** workflow variable we extracted the owner into the **Execute SQL** action.

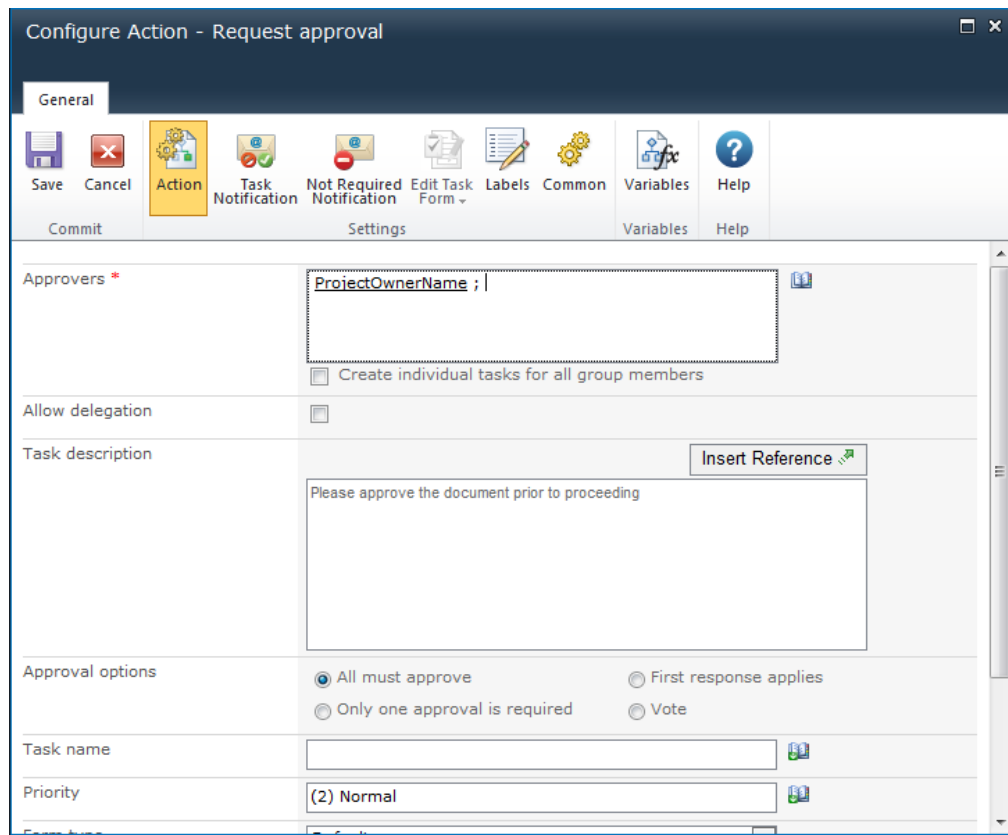


Figure 9: Configure Request Approval Action

Finally, add actions to the approval outcome branches to:

- Notify the submitter if the document approval was rejected.
- Update the security on the document, should it be approved.

- Add a **Send notification** action to the rejected branch

- On the approved branch add a **Set item permissions** action. The resulting workflow should look as outlined in **Figure 10: Completed Approval Workflow**.

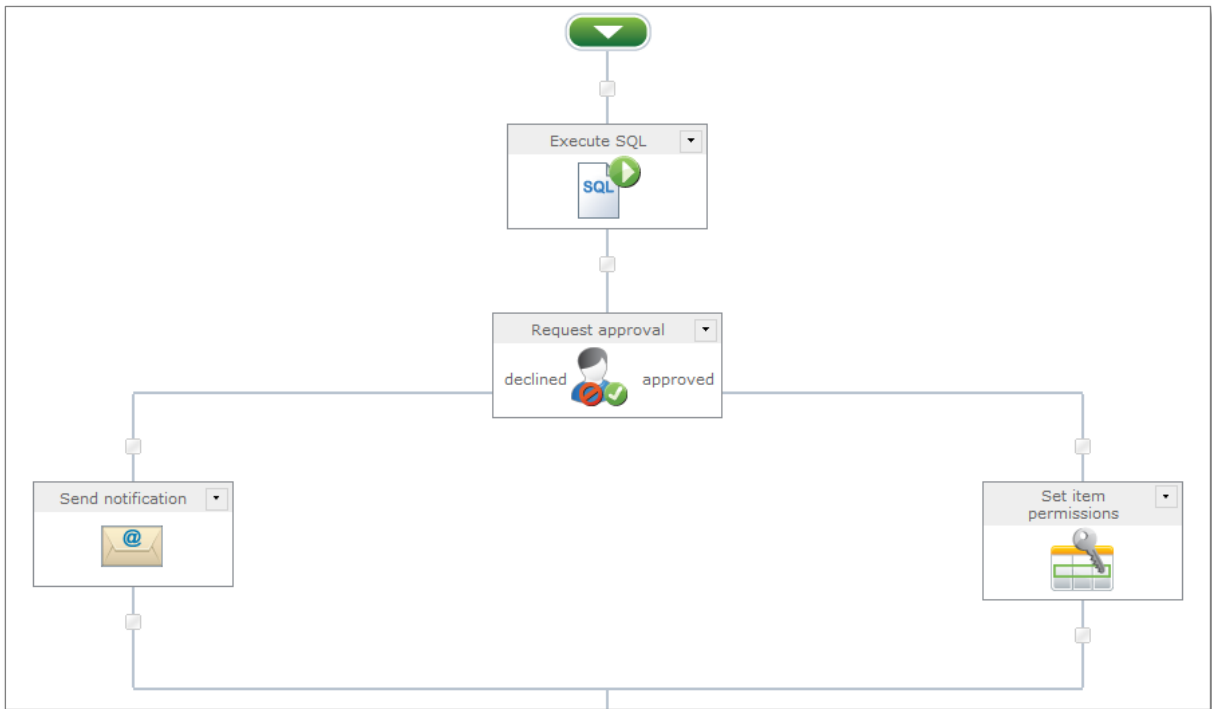


Figure 10: Completed Approval Workflow

9. Add a **Send Notification** action to the declined branch and configure it as follows:
 - a. In the **To** field, click on the lookup icon and select the common workflow variable 'Initiator'
 - b. In the **From** field select the **ProjectManagerOwner** variable.
 - c. In the **Subject** field, enter the text 'Document review – Rejected'
 - d. In the **Body**, enter the following text 'Sorry, but your document was rejected', click on **Insert Reference** and choose to insert the **All Approver Comments (HTML)** variable.
 - e. Click on **Save**.

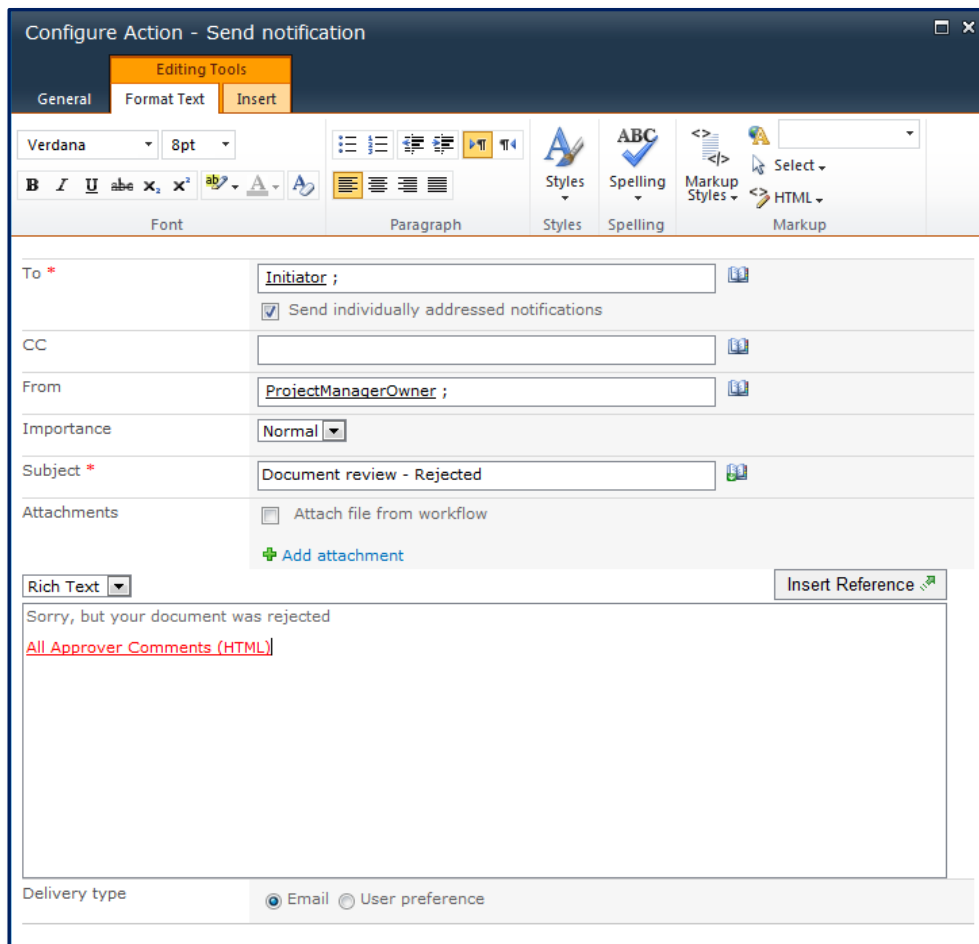


Figure 11: Send Notification Action

10. Double-click on the **Set Item Permissions** action and configure it as follows:
 - a. Verify that the **Set Permission On** field is set to **Current Item**
 - b. Select **No** in the **Inherit permissions from parent** option
 - c. Ensure the **Remove existing permissions** option is **checked**
 - d. We need to set up two sets of **User Permissions** as outlined below, to add a second set of users click on the **Add user permission** link.
 - Set the permissions of the **Project Server Team Members** and **Readers group** to be **Read only**.
 - Set the permissions of the **Project Managers** and **Web Administrators** to be **Full Control**.
 - e. Click on **Save**.

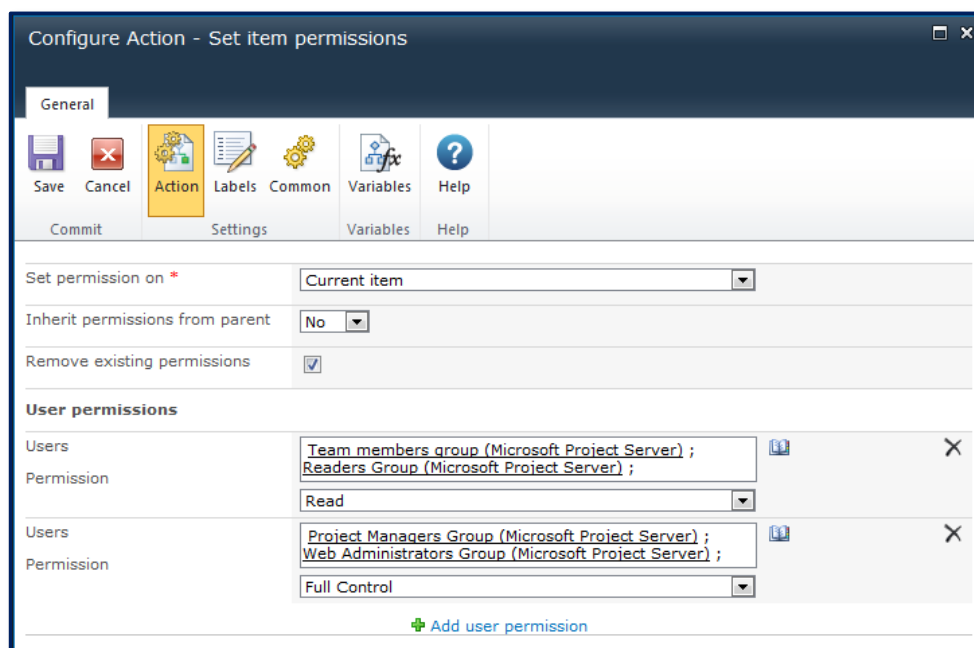


Figure 12: Set Item Permissions

Once published, the workflow will automatically add a workflow approval status column to the list which will be updated by the workflow process depending on the outcome of the approval.

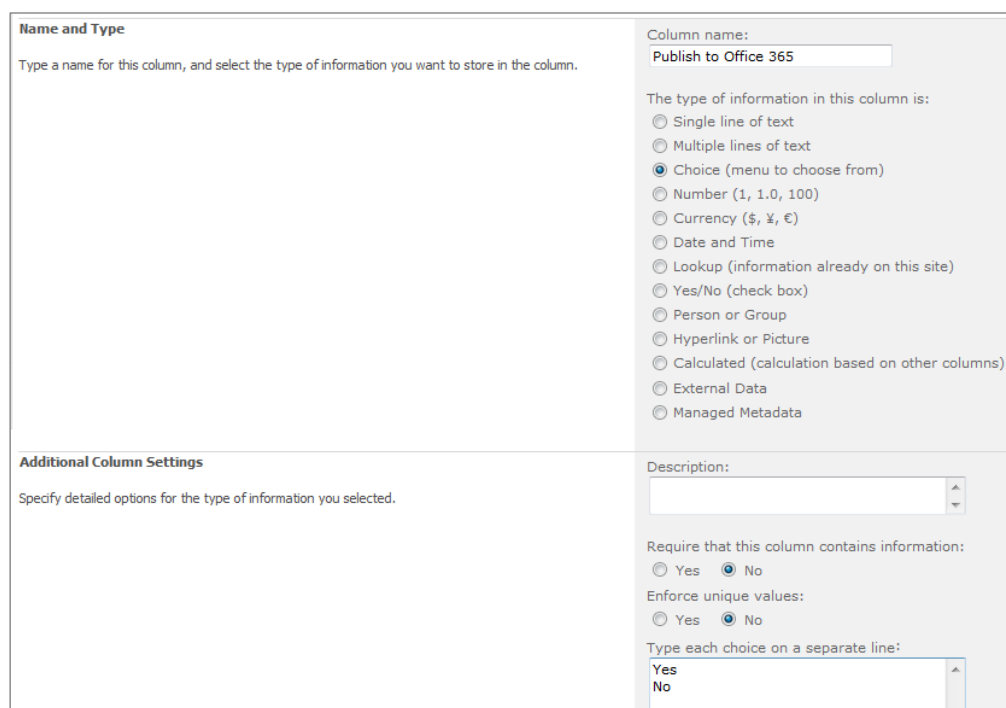
Synchronize Documents with Office 365

With the introduction of Office 365, more and more organizations are leveraging cloud hosted SharePoint instances for collaboration with external parties without needing to provide access to their corporate network infrastructure. An area where this is increasingly common is around organizations which have entered into a joint venture to deliver a project, or are leveraging external contractors or service providers.

In this example, we are going to use a site workflow to synchronize documents from a Project Server workspace within our corporate network to an Office 365 SharePoint site. The workflow will look for documents that have the 'Publish Externally' flag set to 'Yes' and then push them up to Office 365. The process will be repeated each hour by scheduling the site workflow.

Note: Before you begin, ensure Nintex Live has been enabled and configured within your farm.

1. In the Project Workspace, modify the Project Documents library to include a new choice column called **Publish to Office 365** with the values 'Yes' and 'No'.



The screenshot shows the 'Name and Type' dialog for creating a new column. The 'Column name' is 'Publish to Office 365'. The 'Type of information in this column is' is set to 'Choice (menu to choose from)'. The 'Additional Column Settings' section shows 'Require that this column contains information' set to 'No', 'Enforce unique values' set to 'No', and 'Type each choice on a separate line' checked. The choices are 'Yes' and 'No'.

Figure 13: Publish to Office 365 Column

This column will be used to determine if the document will be published to the Office 365 SharePoint site.

2. In this example, we shall upload all relevant documents to the Office 365 SharePoint site on an hourly basis; therefore it is necessary to use a Nintex Site Workflow which can be scheduled.
 - a. From the **Site Actions** menu, select **Create Site Workflow**.

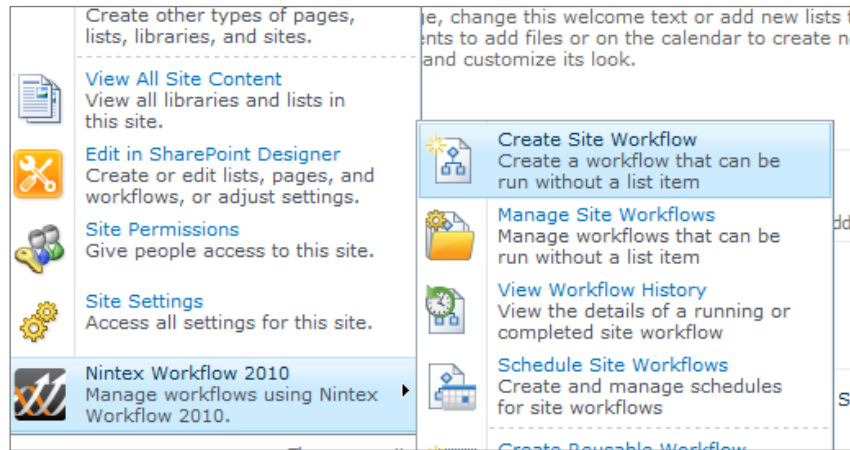


Figure 14: Create Site Workflow

- b. Once the workflow designer has opened, click on the **Nintex Live Catalog** button in the ribbon.

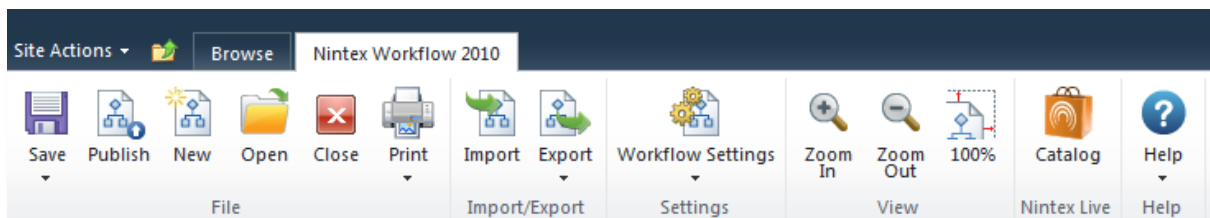


Figure 15: Nintex Live Catalog button

- c. This will open up the catalog of all available Nintex Live actions. Search for Office 365 and click **OK** to add the **Office 365 – SharePoint upload file** action to the Nintex Live category in the Workflow Action menu.

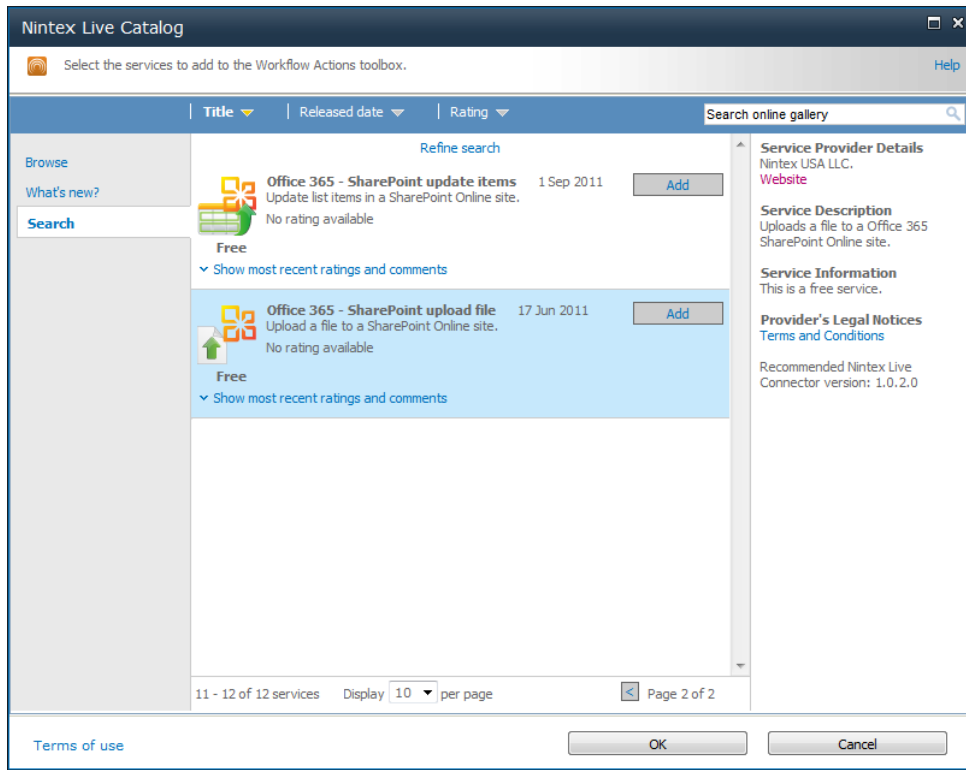


Figure 16: Nintex Live Catalog

3. Add a **Query list** action onto the design canvas, this will be used to query the Project Documents document library and determine which documents have the 'Publish to Office 365' flag set to 'Yes'. Configure the action as follows:
 - a. Ensure the **Editor mode** is set to **Query Builder**
 - b. Select '**Project Documents**' in the **List menu**
 - c. Expand the **filter** area and choose '**Select items only when the following is true:**'
 - d. In the **Show the items when column** section, choose '**Publish to Office 365; Is equal to; Yes;**
 - e. In the **Field** section, select **ID** and click on **Add**
 - f. Choose to save the **ID** field in a new collection variable called **DocUploadCollection**.

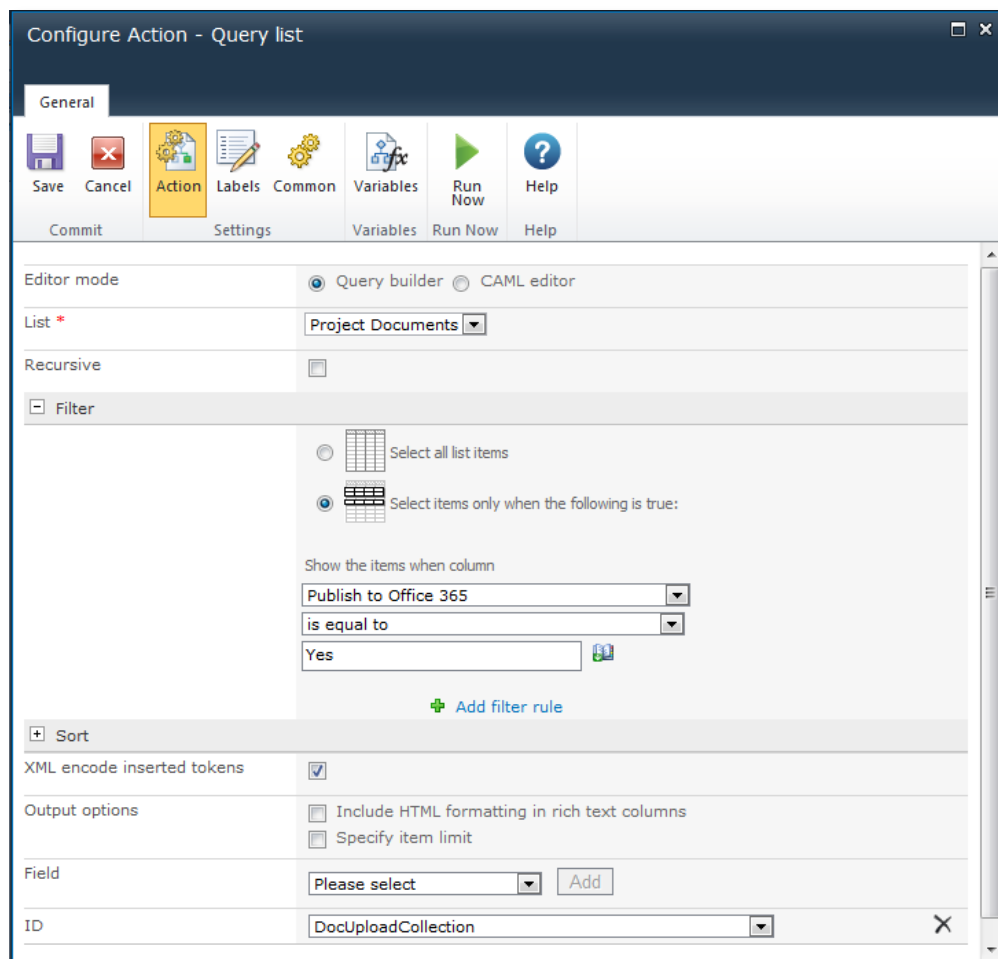


Figure 17: Query Project Document list

4. Add a **For each** action on to the design canvas, this will be used to iterate through the **DocUploadCollection** variable. Configure the **For each** action as follows:
 - a. In the **Target Collection** field, select **DocUploadCollection**
 - b. In the **Store result in** field, create a new **List Item Id** variable called **DocumentToUpload**.
 - c. Click on **Save**.

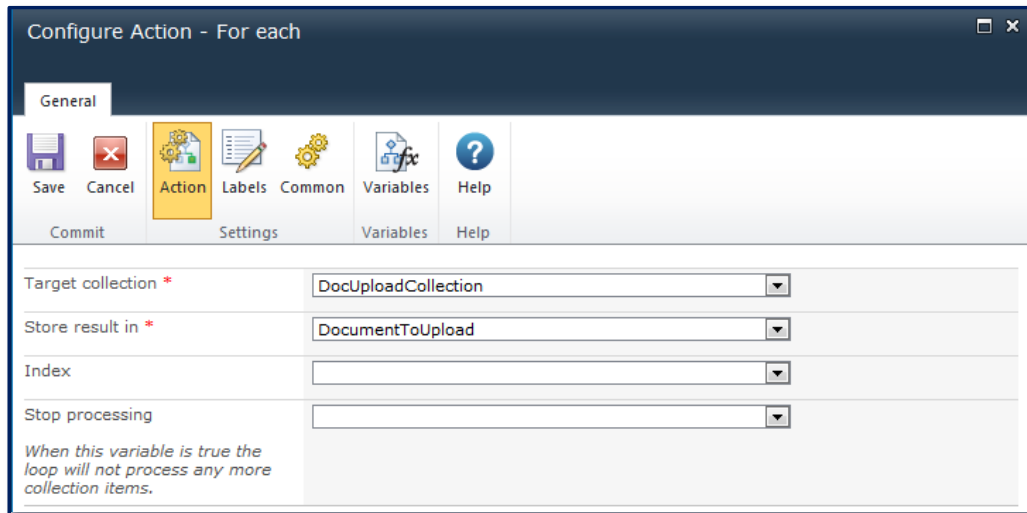


Figure 18: Configure For Each Action

Before the document can be uploaded to Office 365, it is necessary to retrieve some extra data about the document that the **Office 365 –SharePoint upload file** action requires.

5. Add a second **Query list** action within the **For each** action loop.

6. Configure the **Query list** action as follows:
 - a. Ensure the **Editor mode** is set to **Query Builder**
 - b. Select **'Project Documents'** in the **List menu**
 - c. Expand the **filter** area and choose **'Select items only when the following is true'**
 - d. In the **Show the items when column** section, choose **'Id; Is equal to; DocumentToUpload;**
 - e. In the **Field** section:
 - i. select **Name** and click on **Add**
 - ii. select **Absolute Encoded Url** and click on **Add**
 - f. Choose to store:
 - i. the **Name** field in a new single line of text variable called **DocumentName**.
 - ii. the **Absolute Encoded Url** field in a new single line of text variable called **AbsoluteEncodedUrl**
 - g. Click on **Save**

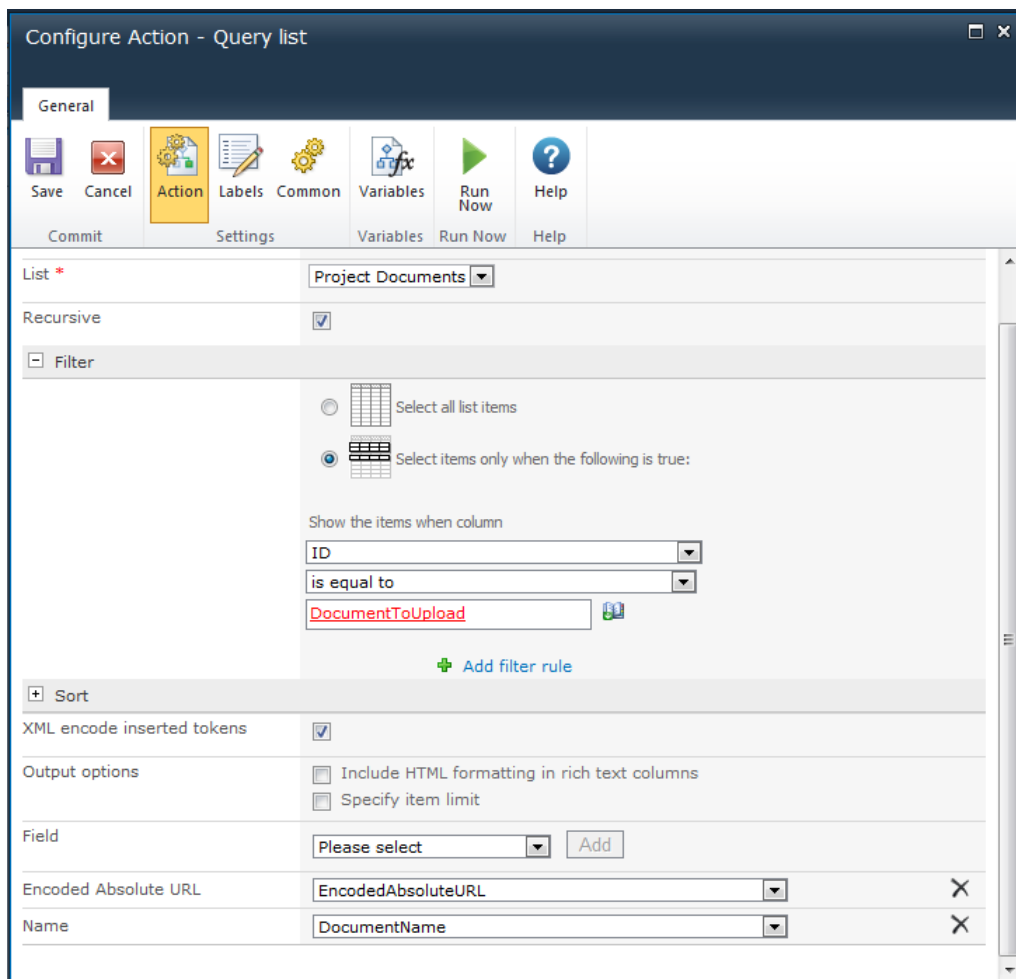


Figure 19: Further Documentation Query List Configuration

7. Once the document information has been retrieved, it is possible to upload the document to the Office 365 SharePoint Site. Drag an **Office 365 – SharePoint upload file** action onto the design canvas and configure it as per below:
 - a. In the File to upload section, select '**Existing Sharepoint content**' and enter the variable **EncodedAbsoluteUrl** in the **URL** field.
 - b. In the **Destination site url** field enter the URL of an Office 365 site you have access to
 - c. In the **Folder Path** field, enter the destination document library to upload your document to, in this case we are using the **Shared Documents** library.
 - d. In the **File name** field, enter the **DocumentName** variable.
 - e. Check the **Overwrite existing file** option.
 - f. In the credentials section, enter the URL of your Office 365 SharePoint tenant and your username and password.
 - g. Click **Save**.

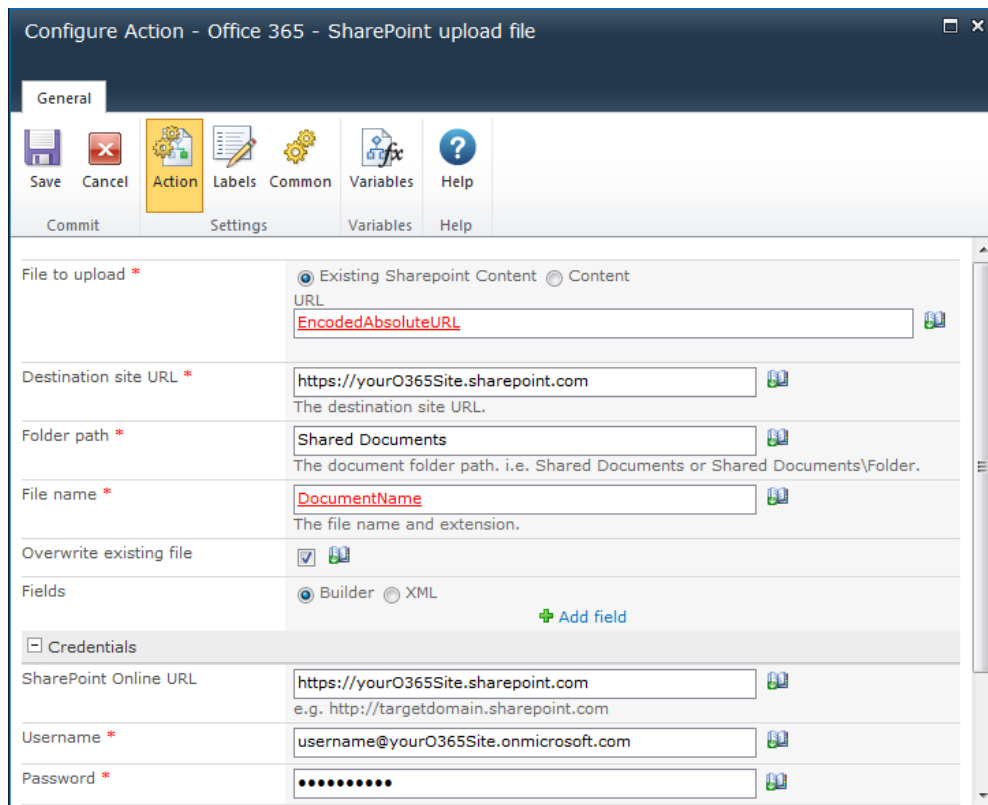


Figure 20: Office 365 - SharePoint Upload File Action Configuration

Once completed, the workflow should look as outline in Figure 21: Completed Office 365 Sync Workflow.

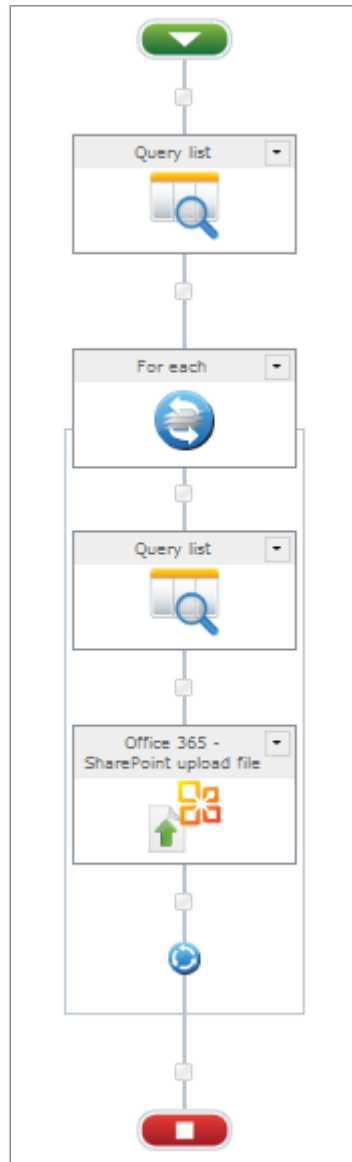


Figure 21: Completed Office 365 Sync Workflow

8. Finally, publish the workflow by clicking on the **Publish** button in the ribbon.

Scheduling the Workflow

As the workflow is designed to be run on an hourly basis, it is necessary to set up a workflow schedule. To do so, select **Site Actions > Nintex Workflow 2010 > Schedule Site Workflows**.

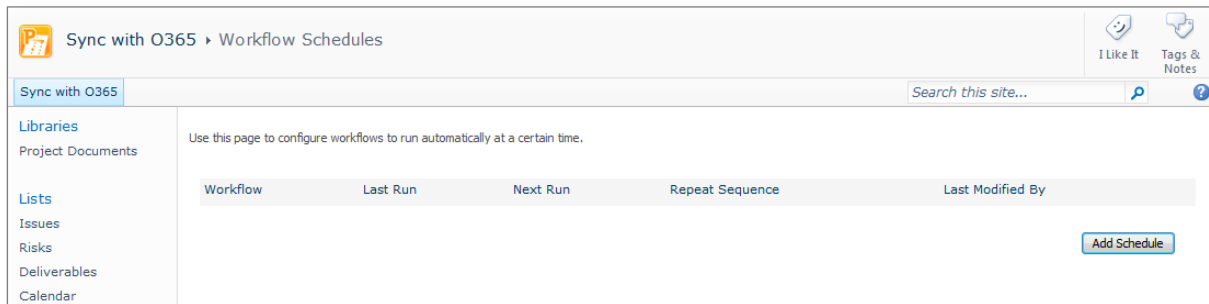


Figure 22: Workflow Schedules

1. Click on **Add Schedule** to configure the settings for the scheduled workflow.

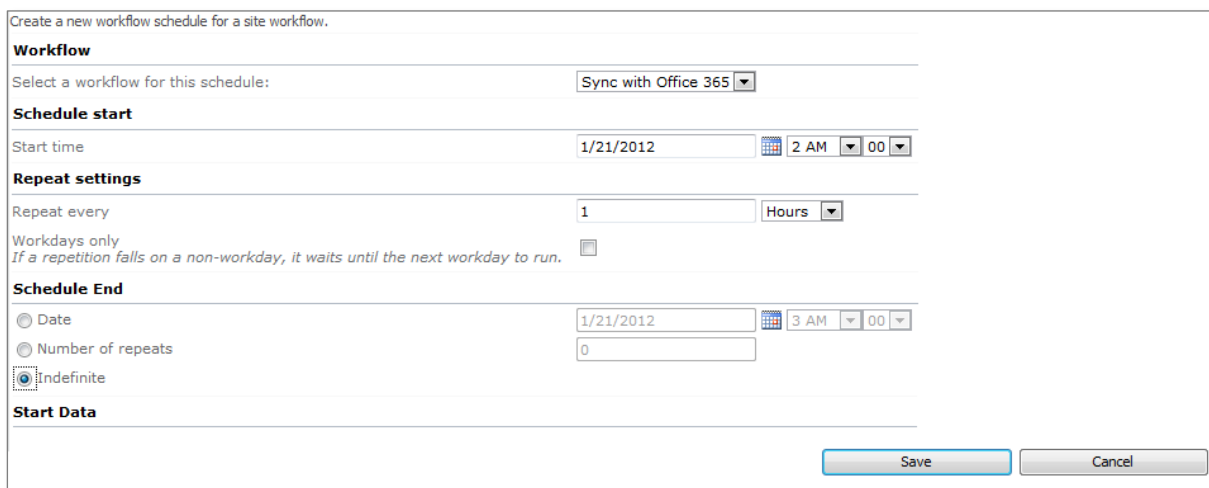


Figure 23: Schedule a Site Workflow

For this example, the schedule is set to repeat every hour indefinitely. To complete the workflow scheduling, click on **Save**. Once configured, the workflow will begin every hour on the hour and upload any documents where the **Publish to Office 365** indicator is set to 'Yes'.

Extending the Example

The example outlined above is very simple, but demonstrates the potential of using Nintex Live with Nintex Workflow 2010 to publish items to an Office 365 hosted SharePoint site. With a little more work, you can extend the workflow to synchronize documents automatically based on the last modified date both to and from Office 365, or synchronize other items that may reside in a workspace, such as risks or issues. Nintex Live provides a rich set of Office 365 actions that can be leveraged along with the out-of-the-box actions to build powerful hybrid Project Server workspace/Cloud hosted solutions for any project.

Conclusion

Nintex Workflow for Project Server 2010 empowers Project and Portfolio Managers to design and implement end-to-end workflows without the need for complex and costly development cycles.

In this final part of the guide we showed how in addition to Demand Management and Event Driven Workflows Nintex Workflow for Project Server leverages the underlying Nintex Workflow integration with SharePoint to automate SharePoint workflows improving collaboration and communication across the project team during project execution and management.

Nintex Workflow for Project Server 2010 helps you to execute your projects on time, within budget, aligned to your corporate strategies.

- **Download a free 30 Day trial now at: www.nintex.com/project.**