SharePoint Workflow Technical Design

Contents

[Author’s Notes 2](#_Toc388344834)

[Use 2](#_Toc388344835)

[Assumptions and Dependencies 2](#_Toc388344836)

[Prerequisites 2](#_Toc388344837)

[Workflow Design 2](#_Toc388344838)

[Workflow Settings 3](#_Toc388344839)

[Workflow Forms 3](#_Toc388344840)

[Initiation Forms 3](#_Toc388344841)

[Task Form 3](#_Toc388344842)

[Association Columns 3](#_Toc388344843)

[Workflow Steps 3](#_Toc388344844)

[Actions 4](#_Toc388344845)

[Task Process Details 4](#_Toc388344846)

[General Settings 6](#_Toc388344847)

[Advanced Settings 6](#_Toc388344848)

[Workflow Notifications 6](#_Toc388344849)

[Confirmation Sent to Creator 7](#_Toc388344850)

[Approval notification sent to Approvers 7](#_Toc388344851)

[Declined 7](#_Toc388344852)

[Deployment Process 7](#_Toc388344853)

[Save Workflow as a Template 7](#_Toc388344854)

[Deploy 7](#_Toc388344855)

[Deploy via Powershell 7](#_Toc388344856)

[Publish Globally 8](#_Toc388344857)

[Apply to Content Type 8](#_Toc388344858)

# Author’s Notes

This document is to be used as a template for creating a SharePoint Designer workflow technical design. It applies to task processes. If you’re not familiar with task processes read Kim Frehe's essential reading [SharePoint Designer 2010 Workflow Advanced Properties](http://community.rightpoint.com/blogs/viewpoint/archive/2012/04/17/sharepoint-designer-2010-workflow-advanced-properties.aspx) and [SharePoint Designer Workflow 2010 Task Actions](http://community.rightpoint.com/blogs/viewpoint/archive/2012/07/20/sharepoint-designer-workflow-2010-task-actions.aspx)

I've done my best to explain what each section is for and what goes in there in the hopes that you can fill in the requirements between the lines. I’ve left some examples for you in each section. Each example in the document is in italics.

Questions and comments can be [emailed to me](mailto:eabderhalden@rightpoint.com?subject=SharePoint Workflow Technical Design).

Happy Technical Designing!

* Erik Abderhalden

# Use

*This document provides design and deploy instructions for your SharePoint Designer Workflow.*

# Assumptions and Dependencies

This section defines what content type(s) the workflow runs for.

*This workflow will be designed to work solely with this content type to support the approval request.*

## Prerequisites

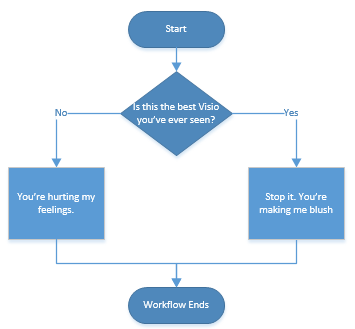
This is fairly straight forward: what needs to be in place so the workflow can run successfully? SharePoint has a lot of moving pieces so it's best that everything that is needed for a workflow to run successfully are captured here. Data connection libraries, columns, content types, and anything else needed can be placed into this section.

*This workflow will use existing site columns from the content type. The content type and site columns will need to be deployed prior to workflow completion in order to use the form fields in the workflow*.

# Workflow Design

This is where the gorgeous Visio process goes. In addition to an image, this section defines what happens in the diagram and where the workflow stores tasks.

*On submit, the form will initiate a SharePoint Designer custom approval workflow as depicted in a gorgeous Visio below. The workflow’s tasks and history will reside in the Department’s default Task and Workflow History lists.*



*Tasks will be assigned to SharePoint Groups. Individuals from the SharePoint groups will claim and complete a task.*

# Workflow Settings

## Workflow Forms

### Initiation Forms

Initiation Forms are the forms that are used to start a workflow. If a custom initiation form is to be used, say so here. If you're running when an item is submitted, this doesn't apply to you.

*This workflow does not require an initiation form.*

OR

*This workflow will utilize a custom initiation form where initiator's can define a due date, and priority for the workflow.*

### Task Form

The task form is the field that users will use to take action on their tasks. If custom fields are to appear on the form, define which fields appear here; otherwise, assume standard task form.

*All fields are read only. User's responding to a workflow task will be able to link to the item in the form.*

### Association Columns

If the workflow hooks up to any columns that reside in the list and may not be on the item's content type, define them here.

## Workflow Steps

This is where the fun starts and the design get very prescriptive.

### Actions

This is where you define the actions for the workflow. The first few steps in the section are global variables that the workflow will use. You can get as fancy here as you'd like or keep it as simple as you'd like. The last step, Start Task Process on Current Item with is the most important step as it kicks off the task process.

1. *Set Variable1 to <value>*
2. *Set variable 2 to <value>*
3. *If Field equals True*
   1. *Set Variable 3 to <Value>*
   2. *Else*
   3. *Set Variable 3 to <Value>*
4. *Start <Task Process Name> on Current Item with <Approvers>*
   * Task Process Form Settings

* *Participants: Approvers*
* *All at once: Parallel*
* *CC: Parameter: cc*
* *Title: Blank*
* *Instructions: Blank*
* *Duration Per Task: Parameter: Duration Per Task*
* *Due Date for Task Process: SharePoint Workflow Approval Process*

### Task Process Details

#### Change the behavior of a single task

##### Before a Task is Assigned

This section is pretty self-explanatory; before a task is assigned, do this. When using dynamic due dates, I usually define them in here. Additionally fields that appear on the task form should be set here as well.

*Set Variable Today to Today's Date*

##### When a Task is Pending

This section defines what happens when a task process assigns a task to a user. The most common step is the following.

*Email task notification to: Current Task: Assigned To*

##### When a Task Expires

This section defines the task process' behavior when a due date for the current task expires.

*Email task reminder notification to: Current Task: Assigned To*

##### When a Task is Deleted

This section is for when a task is deleted. In my experience, I haven't seen a lot of people finagling with their task lists. You can leave this section blank, but for your tech design's sake, you can say:

*Do nothing. This will avoid notifications being sent when the task is rescinded*

##### When a Task Completes

After a task completes, the task process should capture and log the outcome.

*If Current Task: Outcome equals Approved*

*Log Task assigned to Current Task: Assigned To was approved*

*Else if Current Task Outcome equals Rejected*

*Set Variable: Is Item Approved to No*

#### Change the completion conditions for this task process

This is the brain center of the workflow. In this section, the criteria for a successful and a failing workflow are contained. Basically it says, if such and such conditions are met, the workflow is a success. But if such and such contains a rejected item, the workflow stops.

*If Task Process Future Task Count equals 0*

*And Task Process Number of Approved is >= 6*

*Set Variable: IsItemApproved to Yes*

*End Task Process Results Number of Rejected >= 1*

*Set Variable: IsItemApproved to No*

*End Task Process*

#### Change the behavior of the overall task process

This section is where actions outside of tasks are defined. These tend to be a little "lighter" in verbiage compared to the rest of the design document as they tend to be a little more cosmetic and on the governance side of the house, but your mileage may vary.

##### When the Task Process Starts

This section is what is to happen once the task process starts. Usually a "receipt" if you will, is sent to the initiator.

*Send Confirmation Email to requester / author*

##### When the Task Process is Running

Once the task process is up and running, the task process' status should be defined.

*Set Workflow Status to In Process*

##### When the Item is Deleted

If the item is deleted mid-task process, what is to happen?

*Not applicable. All members will have custom permission level: contribute, no delete.*

##### When the item is Changed

This section defines what is to happen if the item is changed mid-workflow.

*Not applicable*

##### When the Task Process is Canceled

Task processes can be terminated through the browser. If a user or admin were to terminate a task process, what should happen?

*Send Cancellation Email to requester / author, all approvers.*

#### When the Task Process Completes

The steps defined in completion conditions apply here. Basically if the completion conditions are met, either pass or fail, what should the task process do?

*If Variable: IsItemApproved equals Yes*

*Set workflow status to Approved*

*Email Workflow Context: Initiator*

*Set workflow status to Approved*

*Else if IsItemApproved equals No*

*Set workflow status to rejected*

*Email Workflow Context:Initiator*

### General Settings

These are general workflow settings. I’ve always struggled about where to place both the General Settings section and the Advanced Settings section in this document, but have generally placed them at the end because they've been alluded to or mentioned explicitly elsewhere in the document. If you follow [Kim's post](http://community.rightpoint.com/blogs/viewpoint/archive/2012/04/17/sharepoint-designer-2010-workflow-advanced-properties.aspx), all of these are grouped together. Feel free to move these two sections to your liking.

* *Only allow task recipients and process owners to read and edit workflow tasks: True*
* *Reassignment: True*
* *Change Requests: False*

### Advanced Settings

* *Allow Change Request: No*
* *Allow Reassign: Yes*
* *Delete Tasks on Completion: No*
* *Expand Groups: No (this is required to create one “group” task that will need to be claimed by a group member)*
* *Permission Set: Default*
* *Task Deleted Outcome: Cancelled*
* *Task Process Owner: A SharePoint User or Group.*

## Workflow Notifications

This is a section of its own where all of the communication to participants is stored and documented. To make your life easier, you should set all of your workflow communications to participants as workflow variables in the Actions section before the task process starts. But it's still best to define the messages here.

**Note**: The workflow notification variables in the actions section should be captured in HTML markup.

### Confirmation Sent to Creator

***Subject****: You Created This Item. ID = XX, Status = Submitted*

***Body****: You have created an item that is being approved. Here is a link for your records. You will be notified once the item has been approved or rejected.*

### Approval notification sent to Approvers

***Subject***: *New Item to Approve. ID = XX, Status = Submitted*

***Body***: A new *item has been created and needs to be approved. Here is a link to the item. You may approve or reject. A single approval or rejection will complete this workflow. For any questions or help, contact the support team.*

### Declined

***Subject****: Your Item, ID = XX, has been Declined.*

***Body***: *Dear <UserName>,*

*We regret to inform you that your item has been declined. If you have any questions or concerns please contact <Person who Rejected> and reference ID XX.*

# Deployment Process

The often overlooked piece of any requirements document, especially workflows, is the deployment process. I've [openly](http://community.rightpoint.com/blogs/viewpoint/archive/2013/06/04/the-quasi-definitive-end-to-end-infopath-deployment-guide.aspx) [lamented](http://community.rightpoint.com/blogs/viewpoint/archive/2014/01/09/the-quasi-definitive-access-web-database-deployment-guide.aspx) about how composite solutions don't have a real deployment strategy, but workflows are a little different. Define the deployment strategy here so there are no surprises for you and your client. I’ve provided examples of the vanilla SharePoint workflow deployment process. If you are using a third-party tool to promote between environments, document those steps here.

## Save Workflow as a Template

1. Develop Workflow in Local Environment
2. Publish to local environment
3. Save as Template
4. Go to Site Assets, download .wsp file

## Deploy

These are the manual deployment steps. To do this through Powershell, go to the next section.

1. Go to Site Actions>Site Settings>Galleries>Solutions
2. Upload Solution
3. Activate Solution
4. Activate Site Feature

### Deploy via Powershell

Add-SPUserSolution -LiteralPath <LiteralPath> -Site <SiteURL>

Install-SPUserSolution –Identity <Solution> –Site <SiteURL>

Enable-SPFeature -Identity <Feature> -Url <URL>

## Publish Globally

1. Open the workflow in SharePoint Designer
2. Confirm the fields are wired up correctly
3. Save and Publish the workflow
4. Publish the Workflow Globally

## Apply to Content Type

1. Go to Site Settings, Content Types select content type
2. Add Workflow to content type