OneSpan Sign for Nintex SharePoint (SP2013) v2

User Guide

Last Updated: October 2018

Contents

General 2

About This Document 2

Terminology 2

Using the OneSpan Sign for Nintex SharePoint Actions 2

Create and Send 4

General 4

Advanced 8

Workflow Variables 9

Check Status and Download 10

Download Documents 11

Workflow Variables 12

Best Practices 15

Questions or Issues? 15

https://developer.esignlive.com/forums/ 15

General

This user guide covers how to use ***OneSpan Sign for Nintex*** SharePoint and explains in detail the steps that should be followed to obtain secure signatures within Nintex.

About This Document

Before you begin, ensure you have completed the installation steps found in the ***OneSpan Sign for Nintex SharePoint Installation and Configuration Guide*** found in the downloaded zip file.

Terminology

Document Package or Transaction – An object within OneSpan Sign that holds the content and definitions for all documents and signers. This is the piece that contains all documents that are sent to the signer to complete.

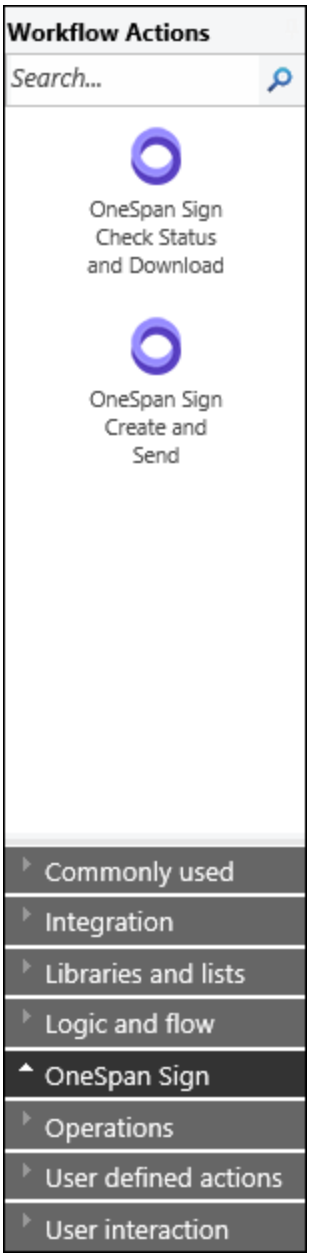
Template – A special type of Document Package/Transaction that can contain documents and signers/signer placeholders and can be used as a starting point when creating a document package/transaction in OneSpan Sign.

Placeholder – When you want to use dynamic signer information (when creating a template), you would use a placeholder signer to allow you to define signature and other field locations for this "signer". When creating a transaction from the template, you should replace the placeholder signer with the actual signer information. All signatures and fields associated with this placeholder will now be assigned to the actual signer.

Using the OneSpan Sign for Nintex SharePoint Actions

1. Create a new workflow inside of Nintex Workflow or open an existing workflow.

2. Within the **Nintex Workflow Designer**, you should see a section in the left pane called ***OneSpan Sign***. If you do not see this, make sure you have installed ***OneSpan Sign for Nintex SharePoint*** properly. Please refer to the ***OneSpan Sign for Nintex SharePoint: Installation and Configuration Guide***

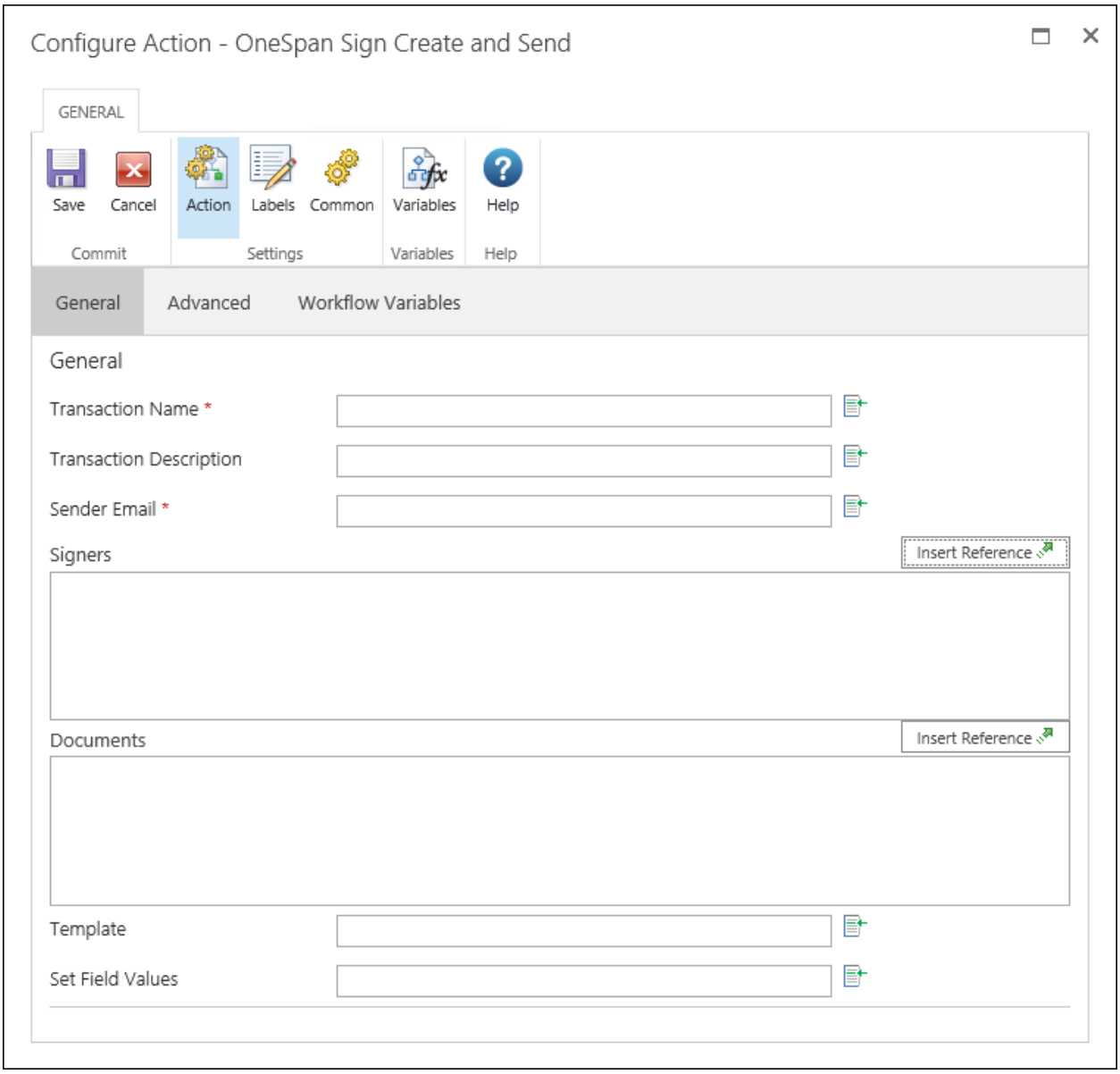

Workflow Actions
Search...
OneSpan Sign Check Status and Download
OneSpan Sign Create and Send
Commonly used
Integration
Libraries and lists
Logic and flow
OneSpan Sign
Operations
User defined actions
User interaction

Create and Send

1. First, you will need to add a ***Create and Send*** action to your workflow.

2. Open the configuration page for the action. You will see three tabs: General, Advanced, and Workflow Variables.

General



Note: All fields have the ability to use the **Insert Reference** option to build dynamic text.

3. First, enter a **Transaction Name** (Example: My Transaction). This is a required field for creating a transaction. **Transaction Description** (Example: This is the description of my transaction.) is an optional field.

4. The **Sender Email** is a required field. In this field, you want to enter the Sender on your OneSpan Sign account that will be the Owner of the transaction. You MUST ensure that the email address you enter in this field is set up as a Sender within your OneSpan Sign account.

5. For **Signers**, if you're using a template with static signers, you will not be required to add anything. However, if you're not using a template, you will need to specify signers for the transaction. Below is the format of what must be entered. Note: This is for a two signer example.

   [{"id":"mysignerid", "name":"mysignername", "emailMessage":"", "index":"0", "title":"Manager", "company":"Acme Inc", "firstName":"John", "lastName":"Smith", "email":"johnsmith@mailinator.com", "authScheme":"SMS", "authChallenge":"+15555555555", "reassign":"true"}, {"id":"mysignerid2", "name":"mysignername", "emailMessage":"Some email message to send just to Jane", "index":"0", "title":"Manager", "company":"Acme Inc", "firstName":"Jane", "lastName":"Doe", "email":"janedoe@mailinator.com", "authScheme":"Challenge", "authChallenge":"{\"answer\": \"0\",\"question\": \"What is 2 - 2?\",\"maskInput\": false}", "reassign":"true", "attachments":"{\"name\":\"attachment\", \"description\":\"some description\",\"required\":true}"}]

   The only optional field is the "attachment" field. Below you will find the options available for each field. Where "**Required"** is noted, you must provide a non-empty-string value.

   - **"id"**: This can be any value, unique to each signer. Example Signer1, Signer2, etc. If you are using a template and must replace placeholders, use this field to enter the placeholder name specified in the template. **Required.**
   - **"name"**: This can be any value and need not be unique. This field can be left empty.
   - **"emailMessage"**: This is a message that will override the transaction level email message found in the **Advanced** tab of this action. This message will only go to the signer it is defined for. This can be an empty string.
   - **"index"**: This value is for the signing order of the signer. If all signers can sign in parallel (at the same time), set this to "0". If you would like to specify an order for your signers, increment this value in the appropriate order, "0", "1", etc.
   - **"title"**: The signer's title which can be used to auto populate a label with the information in the document. This field can be left blank.
   - **"company"**: The company of the signer can be used to auto populate a label in the document. This field can be left blank.
   - **"firstName"**: The signer's first name. **Required.**
   - **"lastName"**: The signer's last name. **Required.**
   - **"email"**: The signer's email address. This must be unique for each signer. **Required.**
   - **"authScheme"**: This is the type of authentication you want the signer to provide to enter the signing ceremony. Options are: "SMS", "CHALLENGE", and "".

- **“authChallenge”**: For SMS, this value would be the phone number, as seen for John Smith, above: “+15555555555”. For CHALLENGE, the value would be the question and answer combination(s) as seen for Jane Doe, above: "{\"answer\": \"0\",\"question\": \"What is 2 - 2?\",\"maskInput\": false}". To add a second question and answer combo, it would look like this: "{\"answer\": \"0\",\"question\": \"What is 2 - 2?\",\"maskInput\": false}, {\"answer\": \"4\",\"question\": \"What is 2 + 2?\",\"maskInput\": false}".
- **“reassign”**: This option can either be “true” or “false” and allows the recipient to specify another person to sign instead of themselves.
- **“attachments”**: This is the only property that doesn’t need to appear in each signer JSON block. As seen in Jane Doe’s block, above, you must specify a name, description, and if the attachment is required. Only description can accept an empty string.

6. For **Documents**, just like **Signers**, if you are using a **Template** that has documents already in it, you will not be required to add anything. If not using a template or if your template contains no documents, you must specify at least one document from SharePoint. The format of the JSON for this field is as follows. Note: this example contains two documents.

   [{"id":"mydocid","name":"mydocname","extractionType":"\"TEXT_TAGS\"","attributes":"\"a1\":\"val1\"","file":"{SharePointSite}/Shared%20Documents/doc1.docx","viewers":"\"mysignerid\""},{"id":"myotherdocid","name":"myotherdocname","extractionType":"\"TEXT_TAGS\",\"ACROFIELDS\"","attributes":"\"a1\":\"val1\",\"a2\":\"val2\"","file":"{SharePointSite}/Shared%20Documents/doc2.pdf","viewers":"\"mysignerid2\""}]

   As with **Signers**, all properties must be specified in each document block. Some may be left blank and will be noted, below. Any fields marked **“Required”** cannot be left as an empty string.

   - **“id”**: This is the id of the document and must be unique to the transaction. **Required.**
   - **“name”**: The name that you want to appear in the signing ceremony for the document. Must be unique within the transaction. **Required.**
   - **“extractionType”**: Documents being drawn directly from SharePoint must use some type of extraction to automatically place fields and signatures within the document. This field specifies the type(s) being used. The options here are \”TEXT_TAGS\” and \”ACROFIELDS\”. You can use both, as you can see above in the second document. **Required.**
     Note: For more information, see Text Tag and Document Extraction guides.
   - **“attributes”**: key/value pairs that can pass through the metadata of the eSL transaction. Value can be an empty string or a comma separated listing of \”key\”:\”value\” pairs.
   - **“file”**: The file that you wish to have signed. **Required.**
   - **“viewers”**: A comma separated list of the IDs of the signers you wish to be able to view a given document. If all can view, pass an empty string (“”).

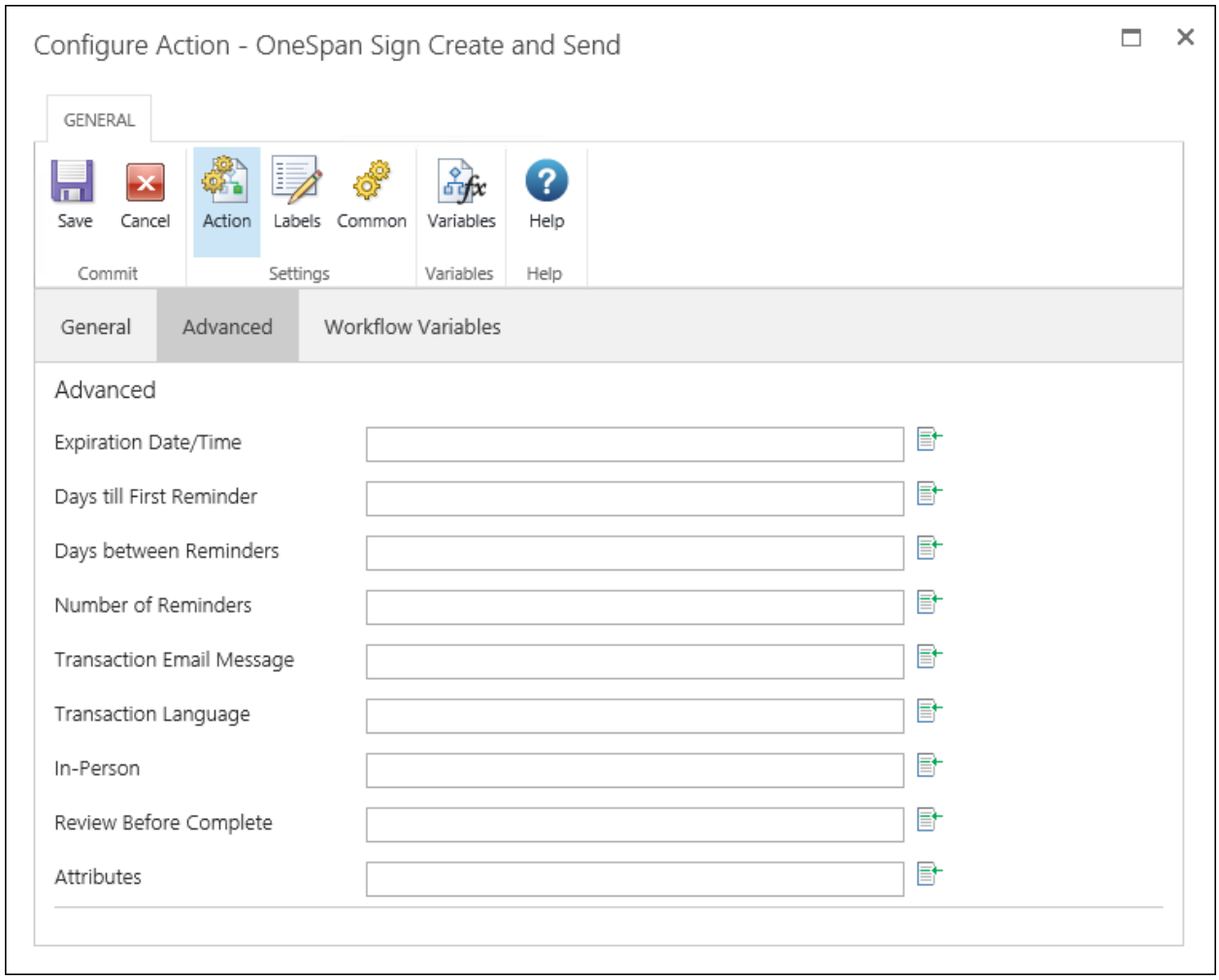
7. For **Template**, you must simply enter the ID of the template. This field is optional and can either be a template to use as the basis for adding more documents from SharePoint or can be used with JUST the template. Example: iAqhRUk21rcFon1Sc2M5BUNx8p4=

8. In **Set Field Values**, you have the ability to prefill your transaction documents with data from SharePoint. The format of this is key/value pairs as follows. Note: This example shows values for fields in two separate documents.

{"mydocid":{"mylist":"value2","mytextarea":"somelongtextvalue","mytextfield":"somevalue","mytextfield2":"someothervalue","mycheckbox":"X","myradiobutton1":"X","myradiobutton2":null,"mylabelfield":"label value"},"myotherdocid":{"mytextfield":"somevalue1","mytextfield2":"someothervalue1"}}

In the above example, you have two documents (mydocid and myotherdocid). In mydocid, you'll see "key":"value" pairs for each field, like: the list "mylist" will be defaulted to "value2", "mycheckbox" will be selected, etc. The key for each field is the field name. You can see or specify a name for each field within a template or within your text tag convention. For more information about text tags, see the guide referenced above in the **extractionType** section.

Advanced



9. **Expiration Date/Time** is the dateTime that you'd like the transaction to expire. This value can be just a date: 2018-05-31. Or it can be a dateTime value: 2018-06-01T08:00:00Z. If you want no expiration date, leave it blank.

10. **Days till First Reminder, Days between Reminders, and Number of Reminders** are integer values that specify a custom reminder schedule. How many days till the signer gets their first reminder, how many days between reminder emails, and finally, the total number of reminders you want them to receive before you stop reminding. These can be left blank.

11. **Transaction Email Message** is a message that will appear in the email received by the signer unless you have overridden this message with a signer level email message, defined in the **Signer** block. This can be left blank.

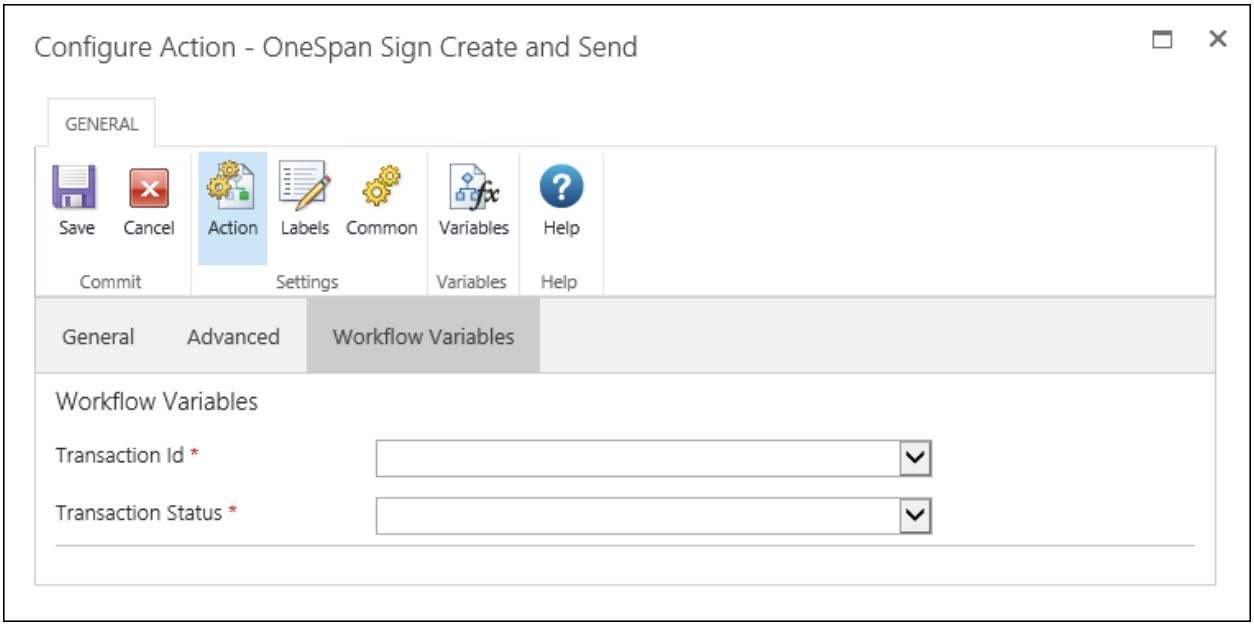
12. **Transaction Language** is the default language of the transaction. Recipients will have the option of choosing their language upon entering the signing ceremony. This can be left blank. The optional values are: en, da, de, es, fr, it, nl, pt, ru, zh-CN, zh-TW, ja, and ko.

13. **In Person** would specify whether the signing ceremony will be happening in person (for example: in the branch of a bank between a banker and a client). Values can be true, false, or blank.

14. **Review Before Complete** can be set to true, false, or blank and specifies if you would like someone to be able to look at the transaction before it is marked complete, in OneSpan Sign.

15. **Attributes** at the package level are much like the ones at the document level described earlier. They are key:value pairs that can pass through the transaction metadata. This can be left blank. An example would be: \"a1\":\"val1\",\"a2\":\"val2\".

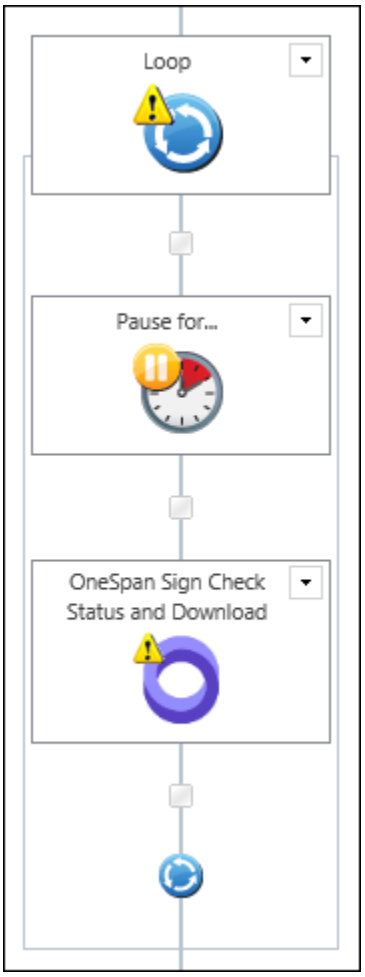
Workflow Variables



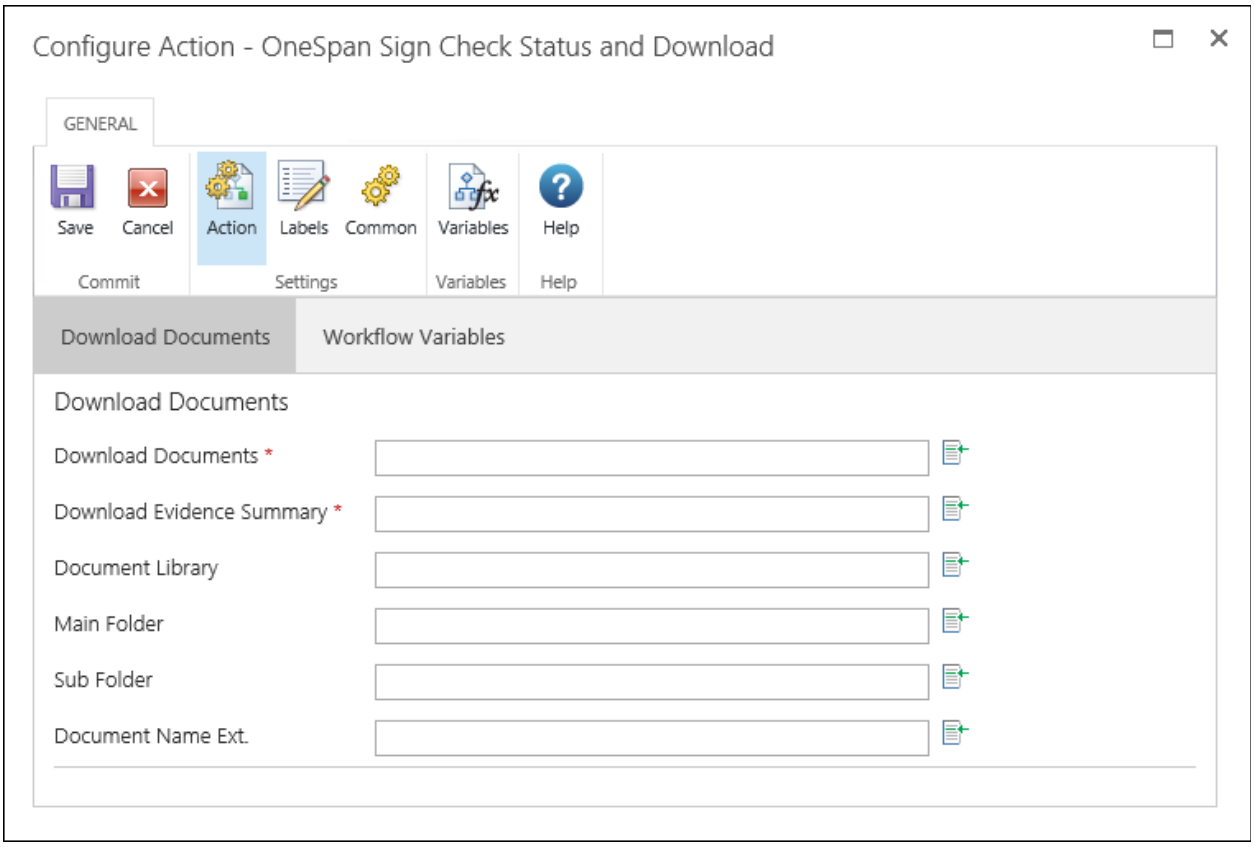
16. **Transaction ID and Transaction Status** are two fields that set the Workflow Variables through which you will pass the Transaction Id and Transaction Status for down-workflow use.

Check Status and Download

1. The next action to add to your workflow will be the **Check Status and Download** action. This simple action will check the status of a transaction and push the status out to a workflow variable. It will also download any specified documents upon completion of the transaction. This action should be surrounded by a **Loop** action from the **Logic and Flow** actions group and should be proceeded by a **Pause for** action found in the **Operations** actions group. Please see the **Best Practices** section for best practices of configuring your workflow to conform to OneSpan Sign's Fair Use policy.



2. Upon entering the configuration of the **Check Status and Download** action, you will see two tabs: Download Documents and Workflow Variables.



Download Documents

3. For the **Download Documents** setting, the values are **true** or **false**. If you choose true, upon completion of the transaction, all documents and required attachments will be downloaded to SharePoint. **Required.**

4. **Download Evidence Summary** is exactly the same as **Download Documents** above. If you choose true, upon completion of the transaction, the evidence summary PDF will be downloaded to SharePoint. **Required.**

5. The **Document Library** setting is only required if either **Download Documents** or **Download Evidence Summary** are set to **true**. This field should point to a Document List in your SharePoint account. For example “Documents”.

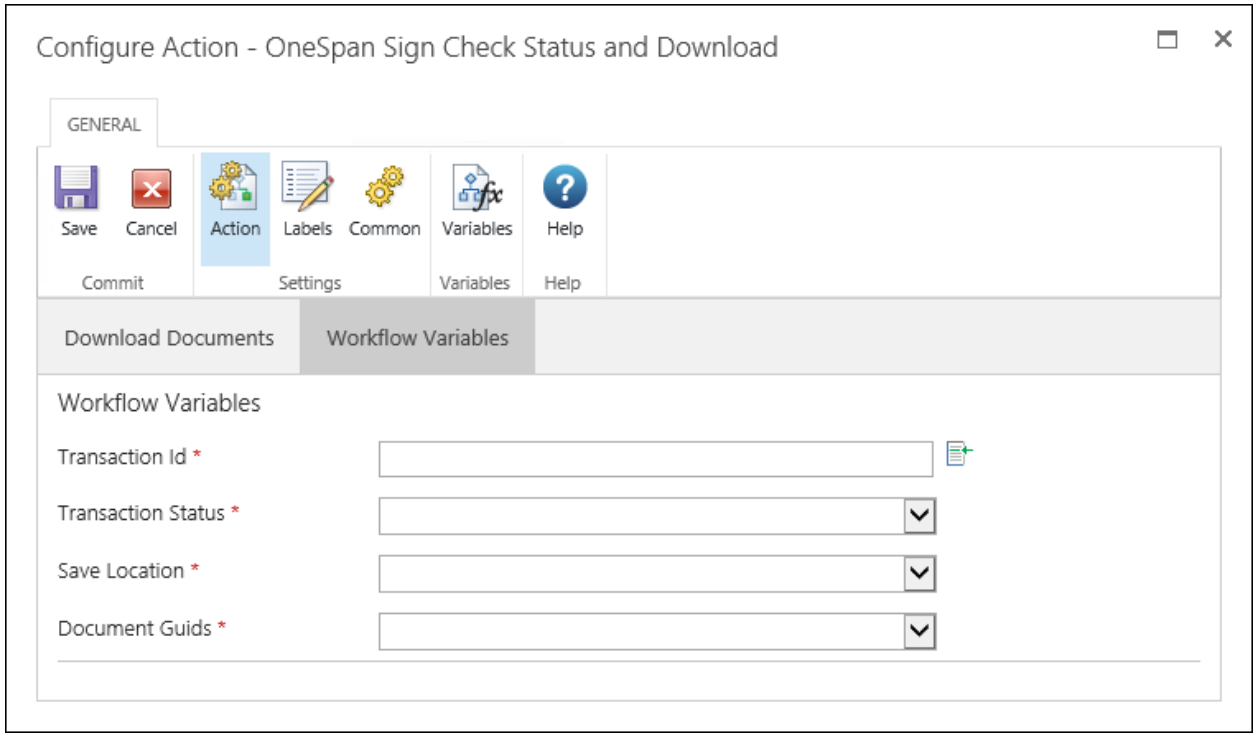
6. **Main Folder** is the first level folder inside the SharePoint Document Library where you would like the files to be saved. For example, “NDAs”.

Note: If you do not choose a value for this but do choose to download any files, the **Transaction Name** will be used for this folder. For example, “Acme Inc. NDA”.

7. **Sub Folder** is the second level folder inside the SharePoint Document Library and inside the **Main Folder** where the files will actually be stored. For example, “Acme Inc”.
   Note: If you do not choose a value for this but do choose to download any files, the **Transaction ID** will be used for this folder.

8. **Document Name Ext.** is an extension that you can add to all files that are downloaded to give them further meaning inside SharePoint. For example, if you chose to save all NDAs into a single folder in SharePoint, you could use this extension to add a company name to all files. For example, “_Acme Inc”

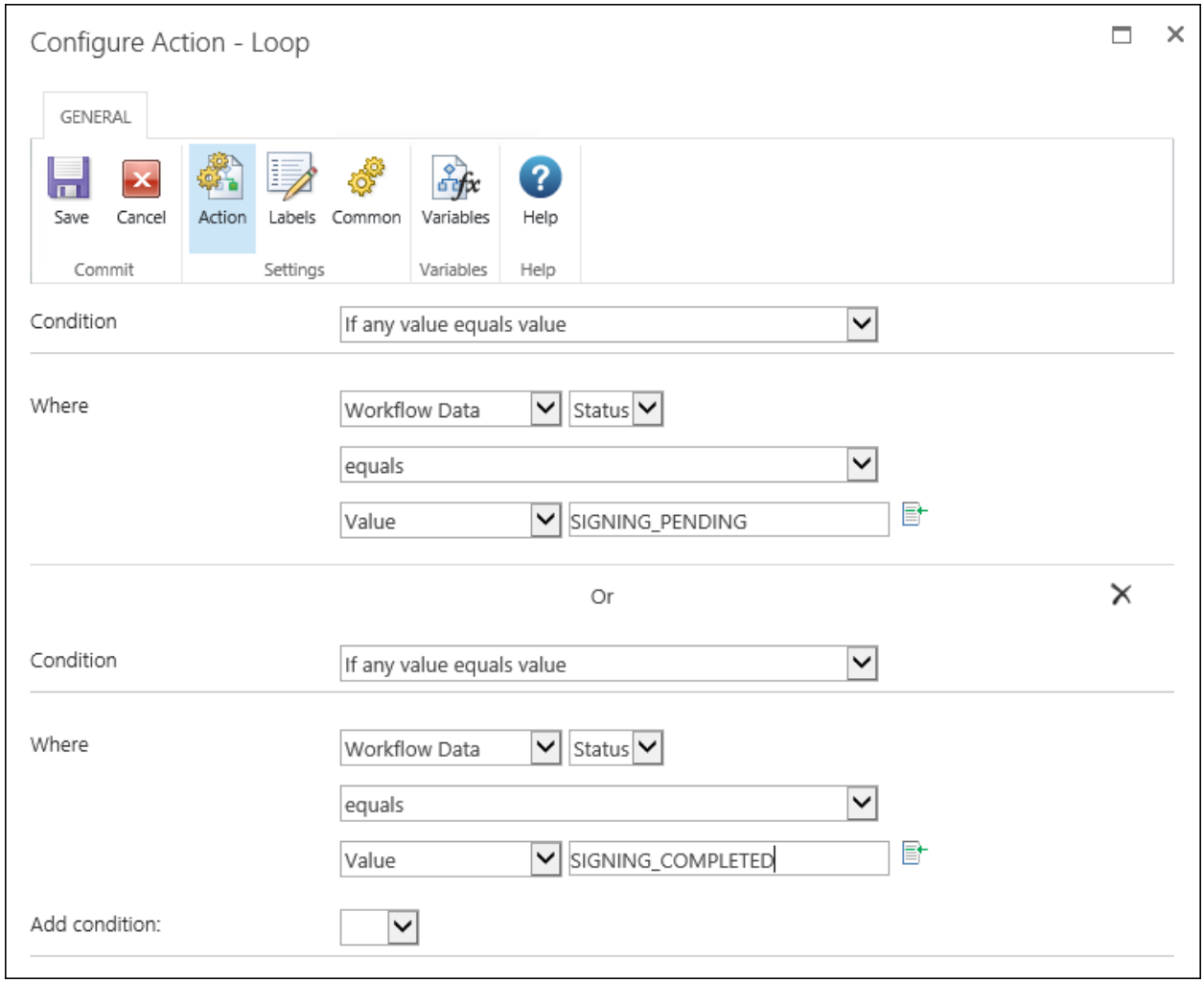
Workflow Variables



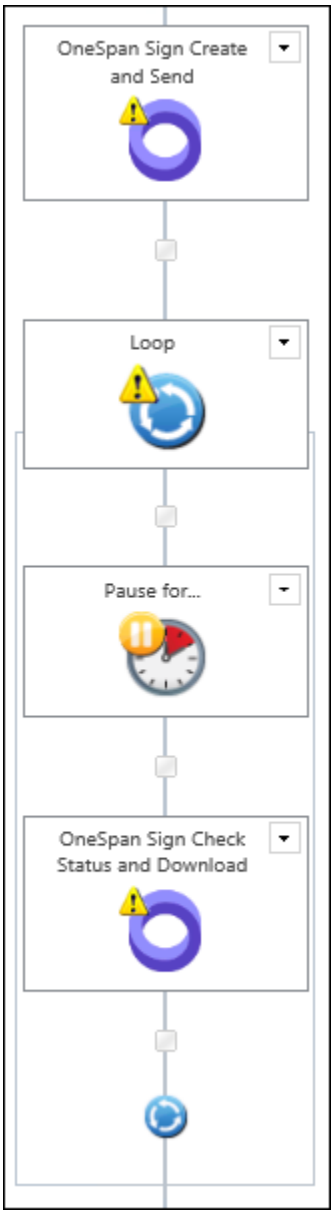
9. The **Transaction Id** variable is the one that was set in the **Create and Send** action. This setting allows the current action to check the status and download documents from the appropriate transaction. The **Transaction Status** variable is the same variable set in the **Create and Send** action, as well. This must be set again, in this action, to allow the **Check Status and Download** action to update the Status variable. **Save Location** will output the save location of the downloaded files. **Document Guids** will output a JSON payload with key:value pairings for documents saved, in the form of “DocumentName”:”DocumentGUID”.

Loop

10. For the **Loop** action surrounding the **Check Status** action, continue to loop as long as the Status workflow variable is still **SIGNING_PENDING**. If you have any required attachments for signers or set **true** for **Review before complete**, you will also need to add an OR condition and check for **SIGNING_COMPLETED** as well. This status appears if signing has been completed, but you're still awaiting review.



11. Now completely configured, the eSignature portion of your workflow should look something like this:



Best Practices

As a best practice for using the Check Status and Download action, please ensure you have a pause step before the action, inside your loop. This pause should be at least an hour (preferably more) long to avoid polling OneSpan Sign for the status of the transaction too often, which could be a violation of the **Fair Use Policy** in your license agreement. If you would like to check the status before an hour, it would be best to configure several **Pause** and **Check Status and Download** actions in series with a check on them for the status variable to avoid downloading documents multiple times. This way, you could pause for 5 minutes, check the status, pause another 15, check the status, pause another 20 and check the status, pause yet another 20, then check the status, then go into a loop where you check every 6-24 hours or something like that. This would allow you to check 4 times in the first hour, in case the process finishes quickly, otherwise, go into a longer holding period if the process does not complete quickly.

Questions or Issues?

Visit the OneSpan Sign Developer Community forums:

https://developer.esignlive.com/forums/