



Accelerate Approval Workflow Package

Version 1.2.0

March 2023

Accelerate Approval Workflow Package

- [Product Info \(Approval Workflow\)](#)
- [Overview \(Approval Workflow\)](#)
- [Admin User Reference \(Approval Workflow\)](#)
- [Technical User Reference \(Approval Workflow\)](#)
- [Troubleshooting \(Approval Workflow\)](#)
- [Release Notes \(Approval Workflow\)](#)
- [Archive of Previous Versions \(Approval Workflow\)](#)

Product Info (Approval Workflow)

Using the Approval Workflow Package you can get full control and track history of your key pricing decisions through automated approval workflows. You can also define their rules based on your needs and utilize them across your key pricing processes - quotes, rebates, price lists, promotions, Live Price Grid items.

| Step | Approval Group | Approval Status | Executed by | Last Action Time | Comment | Reason |
|------------------------|--------------------------|-----------------|--------------------------|------------------|---------|--|
| Margin Alert | Pricing Manager | Approved | Martin Wricke on beh ... | 10/05/2021 13:40 | | The Quote needs to be approved because the margin percent... |
| Approver Total Revenue | Global Pricing Manag ... | Approved | Martin Wricke on beh ... | 10/05/2021 13:40 | | The Quote needs to be approved because the total revenue ... |
| Price Quality | Global Pricing Manag ... | Approved | Martin Wricke on beh ... | 10/05/2021 13:41 | | The Quote needs to be approved because the price quality ... |

Key Deliverables

- Definition of approval workflow types and approval steps for each of them
- Definition of users, user groups or business roles as approvers
- Definition of conditions to apply in each approval step
- Full execution of workflow process
- Workflow history tracking

Workflow Setup

Select workflow type Quote:

Approval Steps

| <input type="checkbox"/> | Step Id | Step Order | Step Label | Reason | Skip | Step Type | Min Appro | |
|--------------------------|---------|------------|-------------------------|-------------------------|------|-----------|-----------|--|
| <input type="checkbox"/> | 1 | 1 | Approve Price | The Quote needs to b... | NO | | | |
| <input type="checkbox"/> | 2 | 2 | Margin Alert | The Quote needs to b... | NO | | | |
| <input type="checkbox"/> | 3 | 3 | Approver Total Reven... | The Quote needs to b... | NO | | | |
| <input type="checkbox"/> | 4 | 4 | Price Quality | The Quote needs to b... | NO | | | |

Add



>> Conditions

Workflow Setup

Select workflow type Quote:

| <input type="checkbox"/> | Step Id | Condition ID | Condition Description | Condition | Skip | |
|--------------------------|---------|--------------|-----------------------|------------------------------------|------|--|
| <input type="checkbox"/> | 1 | 1 | Approve Price | IF (IS_NULL("line.Price"), fals... | NO | |
| <input type="checkbox"/> | 2 | 2 | Margin alert | "line.MarginPct" < 0.2 | NO | |
| <input type="checkbox"/> | 3 | 3 | Total Revenue | "header.Revenue" > 15000 | NO | |
| <input type="checkbox"/> | 4 | 4 | Price Quality | "line.NetPrice" < "line.PriceG... | NO | |

Add



>> Approvers

pricefx

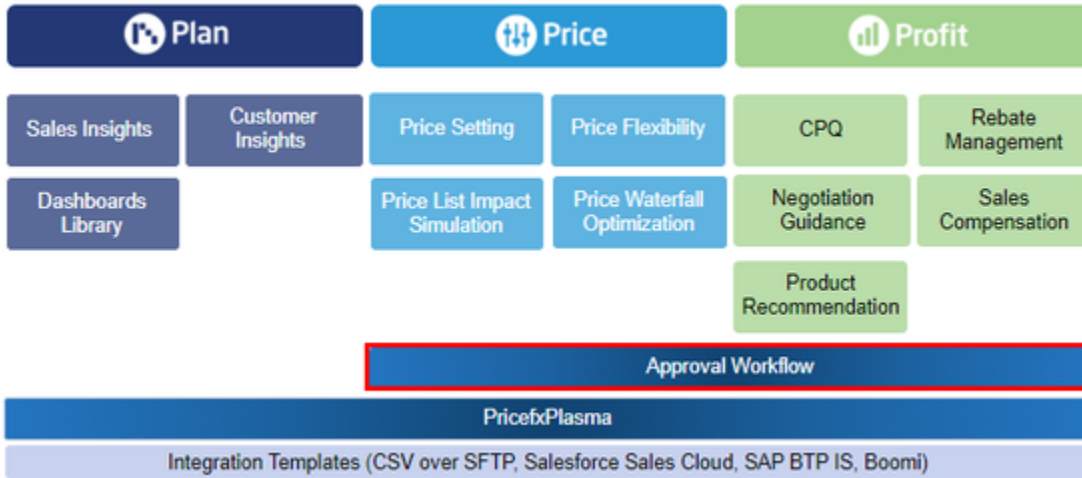
Quote P-714 requires your approval

| | |
|-----------------------------|--|
| Quote Name: | New Quote |
| Created by: | Michal |
| Customer Information | |
| Customer Id: | CD-00004 |
| Customer Name: | Price f(x) AG |
| Country: | Germany |
| Approval Information | |
| Reason | There is a new Quote need to be approved |
| Quote Information | |
| Effective Date: | 2019-09-10 |
| Expiry Date | 2019-09-10 |
| Quote Summary | |
| Total Revenue: | 150 EUR |
| Total Quantity: | 300 |
| Total Margin: | -101 EUR |
| Total Margin %: | -67 % |

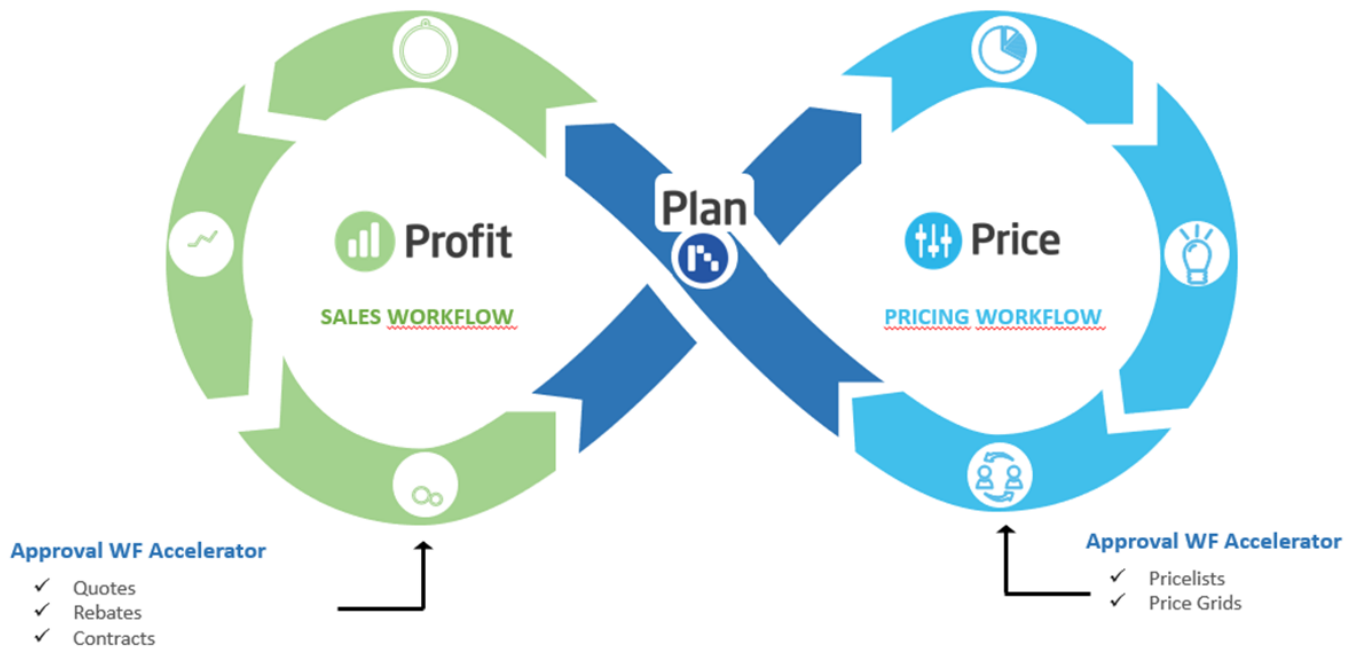
View Quote
Approve
Decline

Overview (Approval Workflow)

The Approval Workflow Accelerator is one of many pre-built solutions from Pricefx that when implemented will provide a customer with a quicker path to the analysis of their pricing data.



Within the design of the Pricefx PLAN/PRICE/PROFIT structure, we can see exactly where the Approval Workflow Accelerator will be included:



Business Overview (Approval Workflow)

Premise

You are involved with pricing applications either in the Pricing, Financial, Sales, or IT support teams within your organization.

Desired Outcome

You need to get full control and track the history of your key pricing decisions through automated approval workflows. Define their rules based on your needs and utilize them across your key pricing processes, like quotes, rebates, price lists, and promotions.

Context and Background

Pricing applications are a core and critical system for many enterprises because the outcomes that they generate can have an oversized impact on margins and profitability. The calculations to generate these outcomes, like price lists, quotes, and rebates, are quite complex and each of them will need an approval workflow to determine if their values are within certain parameters and aligned with corporate goals.

Problem

The workflow approval process is a mandatory part of pricing solutions and invariably its creation can be quite intricate and will almost always require the assistance of an application developer.

Solution Capabilities

Once this Accelerator has been implemented and configured for your environment, these will be the functional capabilities:

- Create custom approval workflows
- Ability to define approval workflow types (i.e. quotes, price list, etc.) and their approval steps

- Link users or user groups as approvers in your workflows
- Create conditional expressions and connect them to the approval step
- Full execution of workflow process
- Workflow history tracking

Capabilities Summary (Approval Workflow)

Capabilities

- Approval workflow is available for following types:
 - Price List
 - Live Price List items
 - Quote
 - Rebate Agreement
 - Rebate Record
 - Promotion Agreements
- Approval workflow can have any number of approval steps.
- Step:
 - Can be conditional:
 - Based on math and logical expression
 - Based on data lookup
 - Can be of type:
 - Approver
 - Watcher
 - Can have one or more:
 - Users
 - User Groups

Audience

- **Technical Admin** - Installs the accelerator and makes the initial setup.
- **Business Admin** (likely Pricing, Sales, or Finance Manager) - Sets up the rules in Company Parameters.
- **Business User** - Uses the system, e.g. submitters, approvers.

User Stories

Approval Level Definition

I want to define the number of approval levels for approvable documents (Quote, Price List, Price Grid, Rebate Agreement, Rebate Records),
so I can create a centralized approval process defined by the roles of my team.

Approver & Watcher Definition

I want to define approvers and/or watchers for each approval level. This can be specific to a user or user group.
So I can empower the appropriate users to review price lists or deals and approve/deny them.

Approval Level Conditions

I **want to** define condition(s) for each approval level,
so I **can** direct quotes / price lists to the right role for their review and approve/reject decision based on their empowerment levels.

Admin User Reference (Approval Workflow)

As an administrator, you can find here detailed descriptions of configuration and customizations of the Approval Workflow Package.

Read on:

- [Installation \(Approval Workflow\)](#)
- [Architecture Components \(Approval Workflow\)](#)

Installation (Approval Workflow)

This installation tutorial is for Approval Workflow Accelerator **version Approval Workflow** and will take you through the use case when you do not have a workflow yet on your partition and you plan to use this Accelerator for approval workflow for your approval process. This tutorial can even be used on a new empty partition.

Pre-requisites


Before you start, ensure that you have:

- **Partition** with the some price list, quote or rebates implementation. Even though you could install on an empty partition, you could not test it.
- [Common Accelerator installation pre-requisites](#)

Installation Steps

Initial Step

1. In Platform Manager, navigate to **Marketplace > Accelerator Packages**.
2. Find *Approval Workflow* accelerator.
3. Click **Detail**, and select your partition and the version (e.g. 1.1.2) you want to install.
4. Click **Deploy**.
5. There is information on the impact of the installation procedure. If you agree with the consequences, go ahead.

 The installation will override previous data tables with the same names as installed by the accelerator and also previous settings.

Sample Workflow Logic

1. Select all the sample logics which you want to install, and then click the **Next** button.
2. If you do not have any workflow logics yet on your partition, it's a good idea to install those samples, as they can be used without any changes as provided.

Sample Workflow Logic

Select Sample Workflow Logic

Select options you would like to use:

Select all Deselect all

Pricelist WF Logic
Pricelist WF Logic

LPG WF Logic
Sample LPG WF Logic

Quote WF Logic
Sample Quote WF Logic

Rebate Agreement WF
Sample Rebate Agreement WF Logic

Rebate Records WF
Sample Rebate Records WF

Next

3. Wait for the deployment of the package. It can take up to several minutes.

Done

The package is installed now. You can click the **Go to partition** button to navigate to the URL of your partition.

If you are curious what the installation process actually installed, review [Architecture of the Approval Workflow accelerator](#).

Post-installation Steps

There are no specific post-installation steps. You can continue with configuration, e.g. with [simple quote configuration](#).

Architecture Components (Approval Workflow)

After installation, you will find the following components deployed at your partition.

Components Shared with Library

Large portion of this accelerator is the same as the Library version. See [Architecture Components \(Approval Workflow Library\)](#) for a list of shared components.

Additional Configurations

Sample Data in Price Parameters

This accelerator contains also initial sample data for the company parameter tables, which are used for configuration of the accelerator.

- ApprovalWorkflowSetup

- Approvers
- ApprovalCondition

To see the sample data, go to [Configuration Tables \(Approval Workflow\)](#).

Sample Workflow Logics

This accelerator can deploy also ready-to-use sample workflow logics, based on your selection during installation process.

- dealplan
- pricegriditem
- pricelist
- quote
- rebatemanagement
- rebaterecord

For more details about those logics see [Workflow Logics \(Approval Workflow\)](#).

Technical User Reference (Approval Workflow)

- [Configure First Simple Workflow for Quote \(Approval Workflow\)](#)
- [Configuration Tables \(Approval Workflow\)](#)
 - [Condition Expressions \(Approval Workflow\)](#)
 - [Values for Condition Expressions \(Approval Workflow\)](#)
 - [Lookup Tables Support in Conditions \(Approval Workflow\)](#)
 - [Add More Approvers to One Step \(Approval Workflow\)](#)
- [Workflow Logics \(Approval Workflow\)](#)
- [Formula Evaluator Library \(Approval Workflow\)](#)

Configure First Simple Workflow for Quote (Approval Workflow)

This tutorial will guide you through a specific simple configuration of approval workflow for a quote. It is expected that you have finished installation of the Approval Workflow accelerator.

At the end, you will have an approval workflow with steps like this:

← New Quote

Recalculate Save **Submit**

Header Items Attachments **Workflow** Messages

| Step | Approval Group | Possible Approvers | Approval Status | Executed by | Last Action Time | Comment | Reason |
|--------------------------|----------------|--------------------|-----------------|-------------|-------------------------|---------|-------------------------------|
| Approve Quote | | Joe Smith | 0/1 | | 06/09/2021 18:24 | | There is a new Quote n |
| Watcher quote | | John Price | Initialized | | 06/09/2021 18:24 | | There is a new Quote ne |
| Supervisor approve quote | Sales Manager | Sales Manager | Initialized | | 06/09/2021 18:24 | | There is a new big Quote |

Configuration Steps

1. Navigate to the company parameter *ApprovalWorkflowSetup* and review the data. Our sample data defined three steps where the second one is of the type *Watcher*. We will not change anything here.

The screenshot shows the 'Price Parameters' configuration page. On the left, a sidebar lists categories like 'Name', 'Label', 'Search...', 'deployedAccelerators', 'ApprovalCondition', 'ApprovalWorkflowSetup', 'Approvers', 'WFExpressionVariableCo...', and 'WFLookupFilterConfigur...'. The 'ApprovalWorkflowSetup' parameter is selected. The main table, titled 'Price Parameter Values: Approval Workflow Setup', contains the following data:

| Workflow Type | Step ID | Step Order | Step Label | Reason | Skip | Step Type |
|---------------|---------|------------|---------------------------|--|------|-----------|
| PriceGridItem | 1 | 1 | Approve for PLI | There is a new PGL must be approve | Yes | |
| PriceList | 1 | 1 | Approve for new PL | There is a new PL, Please approve it | Yes | |
| Quote | 1 | 1 | Approve Quote | There is a new Quote need to be approved | No | |
| Quote | 2 | 2 | Watcher quote | There is a new Quote need to be check | No | Watcher |
| Quote | 3 | 3 | Supervisor approve quote | There is a new big Quote. Please approve it... | No | |
| RebateRecord | 1 | 1 | Approve for Rebate Record | There is a new Rebate must be approve | No | |
| Rebates | 1 | 1 | Approve for Rebate | There is a new Rebate must be approve | No | |

2. Navigate to the company parameter *Approvers*, and modify the sample data so that for all "Quote" records the value of the field Approver Name contains the existing user login and existing group name, otherwise the workflow will fail. In our case, we created two new users with logins "joe.smith" and "john.price" (for this tutorial you only have to set up a login and email for the user), and a new user group named "SalesManager". See also how to [add more approvers for one approval step](#).

The screenshot shows the 'Price Parameters' configuration page with the 'Approvers' parameter selected. The main table, titled 'Price Parameter Values: Approvers at each step', contains the following data:

| Workflow Type | Step ID | Approver ID | Approver Type | Approver Name | Skip |
|---------------|---------|-------------|---------------|----------------|------|
| PriceGridItem | 1 | 1 | USER GROUP | Admin | No |
| PriceList | 1 | 1 | USER GROUP | admin | No |
| Quote | 1 | 1 | USER | joe.smith | No |
| Quote | 2 | 1 | USER | john.price | No |
| Quote | 3 | 1 | USER GROUP | SalesManager | No |
| RebateRecord | 1 | 1 | BUSINESS ROLE | BusinessRole01 | No |
| Rebates | 1 | 1 | USER | vinh | No |

3. Navigate to the company parameter *Approval Conditions* and modify all the "Quote" records to be skipped (set their field **Skip** to Yes), i.e. the conditions will not be used for now.

| Workflow Type | Step ID | Condition ID | Condition Description | Condition | Skip |
|---------------|---------|--------------|-----------------------|--|------|
| PriceGridItem | 1 | 1 | Local Price | (LocalPrice > 1.5) AND ((0.8 + 0.7) <= Adjustment) | No |
| PriceList | 1 | 1 | No condition | | No |
| Quote | 1 | 1 | Total Revenue | header.TotalRevenue > 100 && quoteType == 'CPQ' | Yes |
| Quote | 2 | 1 | MarginPct | header.TotalRevenue > 500 AND line.MarginPctOutput <= 0.55 | Yes |
| Quote | 3 | 1 | Quantity | line.folder.TotalQuantity > 100 AND label == "MB" | Yes |
| RebateRecord | 1 | 1 | Check Rebate | attribute2 >= 1000 | No |
| Rebates | 1 | 1 | Check Target Input | line.Target < 20000 | No |

4. Navigate to **Configuration > Logics > Workflow Logics > General Workflow Logics**. You should see the sample workflow logics provided by the package. Because the name of the quote workflow logic is "quote", the system will use it by default, so you do not even have to register it somewhere. We will not change anything here.

See also how to [develop your own workflow logic](#).

| Workflow Type | Name | Label | Valid After | Status |
|------------------|-----------------|-------------------------|-------------|--------|
| Deal Plan | dealplan | Deal Plan workflow | 01/01/2021 | Active |
| Price Grid Item | pricegriditem | PGI workflow | 01/01/2019 | Active |
| Price List | pricelist | PL workflow | 01/01/2019 | Active |
| Quote | quote | Quote workflow | 01/01/2019 | Active |
| Rebate Agreement | rebateagreement | Rebate Agreement w... | 01/01/2018 | Active |
| Rebate Record | rebaterecord | Rebate Record workfl... | 01/01/2018 | Active |

5. Now make a new Quote of the type "(Default)". Because we switched off all conditions, you do not have to even add any products, simply navigate to the Workflow tab. The workflow logic will be triggered and you should see the approval workflow steps.

| Step | Approval Group | Possible Approvers | Approval Status | Executed by | Last Action Time | Comment | Reason |
|--------------------------|----------------|--------------------|-----------------|-------------|------------------|---------|--------------------------|
| Approve Quote | | Joe Smith | 0/1 | | 06/09/2021 18:24 | | There is a new Quote n |
| Watcher quote | | John Price | Initialized | | 06/09/2021 18:24 | | There is a new Quote ne |
| Supervisor approve quote | Sales Manager | Sales Manager | Initialized | | 06/09/2021 18:24 | | There is a new big Quote |

6. Congratulations! You have made your first simple workflow.

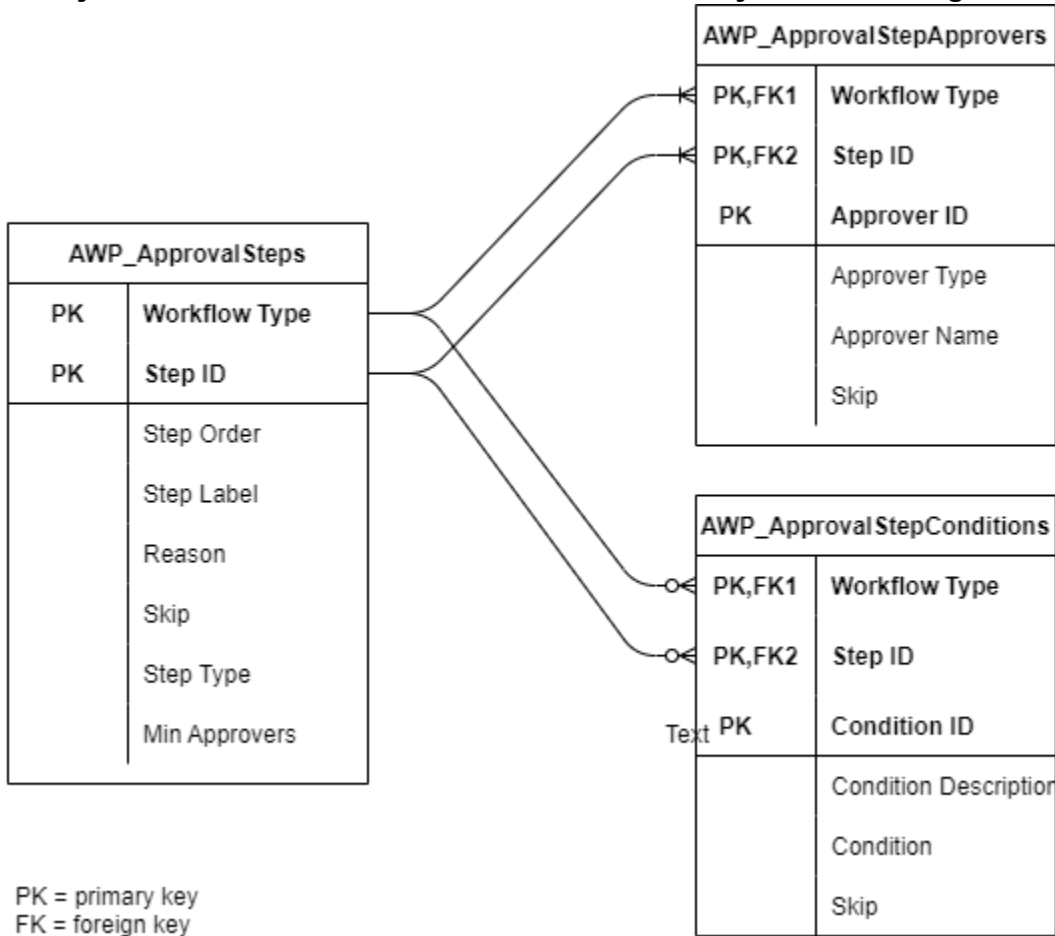
Configuration Tables (Approval Workflow)

There are three price parameters to configure behavior of Approval Workflow Package:

- Approval Workflow Setup - Defines a list of approval steps in the approval workflow for each type.
- Approvers - Defines a list of persons/groups who approve/deny the approval step.

- Approval Conditions - Defines condition under which the the approval step will appear in the approval workflow.

Those three tables are linked together by a unique combination of fields Workflow Type and Step ID. When you write new records to those tables, ensure that you use matching values of those keys.



In this section:

- [Price Parameter "ApprovalWorkflowSetup"](#)
- [Price Parameter "Approvers"](#)
- [Price Parameter "ApprovalConditions"](#)

Price Parameter "ApprovalWorkflowSetup"

Defines a list of approval steps for a given type of document (Workflow Type).

Price Parameter Values: Approval Workflow Setup

...

| <input type="checkbox"/> | Workflow Type | Step ID | Step Order | Step Label | Reason | Skip | Step Type | Min Approvers |
|--------------------------|--|--|--|--|---|--------------------------------------|--------------------------------------|--|
| | <input type="text" value="Search..."/> | <input type="text" value="Search..."/> | <input type="text" value="Search..."/> | <input type="text" value="Search..."/> | <input type="text" value="Search..."/> | <input type="text" value="Sele..."/> | <input type="text" value="Sele..."/> | <input type="text" value="Search..."/> |
| <input type="checkbox"/> | PriceGridItem | 1 | 1 | Approve for PLI | There is a new PGL must be approve | Yes | | |
| <input type="checkbox"/> | PriceList | 1 | 1 | Approve for new PL | There is a new PL, Please approve it | Yes | | |
| <input type="checkbox"/> | Quote | 1 | 1 | Approve Quote | There is a new Quote need to be approved | No | | |
| <input type="checkbox"/> | Quote | 2 | 2 | Watcher quote | There is a new Quote need to be check | No | Watcher | |
| <input type="checkbox"/> | Quote | 3 | 3 | Supervisor approve quote | There is a new big Quote. Please approve it now | No | | 2 |
| <input type="checkbox"/> | RebateRecord | 1 | 1 | Approve for Rebate Record | There is a new Rebate must be approve | No | | |
| <input type="checkbox"/> | Rebates | 1 | 1 | Approve for Rebate | There is a new Rebate must be approve | No | | |

| Column Name | Description |
|---------------|--|
| Workflow Type | <p>Unique name for your approval workflow. You will reference this value from your workflow logic, so this could be any text value of your choice, but generally we recommend to stick with the names suggested in our code samples, like Quote, PriceList, PriceGridItem, Rebates, RebateRecord, DealPlan.</p> <p>Note: If you use different workflows for different Quote Types, use different values for this field - e.g., "Quote1", "Quote2" - and then refer the proper value from your workflow logics.</p> |
| Step ID | <p>Unique ID of the approval step. There is no naming convention/rule for the value of Step ID, you can use any text value. Generally we recommend a number. Steps are sorted by their IDs.</p> |
| Step Order | <p>Order/priority number of the approval step in which they will appear in the approval workflow.</p> |
| Step Label | <p>Label of the approval step. It is displayed to the business user in the list of workflow steps.</p> |
| Reason | <p>Description of the reason why this approval step is needed. This text will be displayed to the creator in the list of approval steps. It is also displayed to the approver, so they understand why they are asked to approve or deny. The text can also be part of the workflow email notifications.</p> |
| Skip | <p>Decides if this step will be used in the approval workflow or not. You can select from two values:</p> <ul style="list-style-type: none"> • "Yes" - This step is not added into the workflow. You can use this value to hide (not use) the step. • "No" (or empty) - This step is added into the workflow. |
| Step Type | <p>Decides if this step is a request for approval, or only a watcher. You can select from two values:</p> <ul style="list-style-type: none"> • "Approval" (or empty) - The step will request some approvers to approve or deny. • "Watcher" - Users selected in this step will only get notification but will not be asked to approve or deny. |
| Min Approvers | <p>How many people must approve this step, so it is considered approved. This setting is valid for approvers with the type GROUP only (approvers are defined in the price parameter Approvers). For other approver types, keep empty.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • (Empty) - Only 1 approver must approve. • Number - Specified number of approvers must approve, so that this step is approved. |

Price Parameter "Approvers"

Price Parameter Values: Approvers at each step



| <input type="checkbox"/> | Workflow Type <input type="text" value="Search..."/> | Step ID <input type="text" value="Sear..."/> | Approver ID <input type="text" value="Search..."/> | Approver Type <input type="text" value="Select Val..."/> | Approver Name <input type="text" value="Search..."/> | Skip <input type="text" value="Select Value"/> |
|--------------------------|---|---|---|---|---|---|
| <input type="checkbox"/> | PriceGridItem | 1 | 1 | USER GROUP | Admin | No |
| <input type="checkbox"/> | PriceList | 1 | 1 | USER GROUP | admin | No |
| <input type="checkbox"/> | Quote | 1 | 1 | USER | test2 | No |
| <input type="checkbox"/> | Quote | 2 | 1 | USER | tram | No |
| <input type="checkbox"/> | Quote | 3 | 1 | USER GROUP | Admin | No |
| <input type="checkbox"/> | RebateRecord | 1 | 1 | BUSINESS ROLE | BusinessRole01 | No |
| <input type="checkbox"/> | Rebates | 1 | 1 | USER | vinh | No |

| Column Name | Description |
|---------------|---|
| Workflow Type | Name of the document/workflow type. This value links the approver to the workflow step, so the value must match a value of Workflow Type in the Price Parameter ApprovalWorkflowSetup. |
| Step ID | ID of the approval step. This value links the approver to the workflow step, so the value must match a value of Step ID in the Price Parameter ApprovalWorkflowSetup. |
| Approver ID | Unique text ID of the approver within the same workflow step. If you have more approvers for one step, ensure you use different values of <i>Approver ID</i> . This value could be any text, but generally we recommend to use numbers. |
| Approver Type | Type of the approver. You can select one of these values: USER, USER GROUP or BUSINESS ROLE. |
| Approver Name | Name of the approver (mandatory value). <ul style="list-style-type: none"> For approver type USER GROUP, enter a name of a user group. For approver type USER, enter a login name. For approver type BUSINESS ROLE, enter a name of a business role. <p>Note: The name of login, group or business role must match exactly the definition in Administration > Access Admin.</p> |
| Skip | Decides if this approver will be used in the workflow step or not. You can select from two values: <ul style="list-style-type: none"> "Yes" - This approver is not added into the workflow step. You can use this value to hide (not use) the approver. "No" (or empty) - This approver is added into the workflow step. |

When you need more approvers for the same approval step, you add more rows to the Price Parameter Approvers.

Price Parameter “ApprovalConditions”

Defines conditions under which the workflow step appears in the workflow.

Price Parameter Values: Approval conditions at each step

+ Add Record ...   

| <input type="checkbox"/> Workflow Type | Step ID | Condition ID | Condition Description | Condition | Skip |
|--|------------------------------|--------------------------------|--------------------------------|--|-----------------------------------|
| Search... <input type="text"/> | Sear... <input type="text"/> | Search... <input type="text"/> | Search... <input type="text"/> | Search... <input type="text"/> | Select Value <input type="text"/> |
| <input type="checkbox"/> PriceGridItem | 1 | 1 | Local Price | (LocalPrice > 1.5) AND ((0.8 + 0.7) <= Adjustment) | No |
| <input type="checkbox"/> PriceList | 1 | 1 | No condition | | No |
| <input type="checkbox"/> Quote | 1 | 1 | Total Revenue | header.TotalRevenue > 100 && quoteType == 'CPQ' | No |
| <input type="checkbox"/> Quote | 2 | 1 | MarginPct | header.TotalRevenue > 500 AND line.MarginPctOutput <= 0.55 | No |
| <input type="checkbox"/> Quote | 3 | 1 | Quantity | line.folder.TotalQuantity > 100 AND label == "MB" | No |
| <input type="checkbox"/> RebateRecord | 1 | 1 | Check Rebate | attribute2 >= 1000 | No |
| <input type="checkbox"/> Rebates | 1 | 1 | Check Target Input | line.Target < 20000 | No |

| Column Name | Description |
|-----------------------|--|
| Workflow Type | Name of the document/workflow type. This value links the approver to the workflow step, so the value must match the value of Workflow Type in the Price Parameter ApprovalWorkflowSetup. |
| Step ID | ID of the approval step. This value links the approver to the workflow step, so the value must match a value of Step ID in the Price Parameter ApprovalWorkflowSetup. |
| Condition ID | Unique text ID of the condition within the same workflow step. If you have more conditions for one step, ensure you use different values here. This value could be any text, but generally we recommend to use numbers. |
| Condition Description | Text describing the condition. It helps you understand what the condition does. |
| Condition | <p>Boolean expression which is evaluated to determine if the workflow step should be added to the workflow or not.</p> <p>The workflow step is added to the workflow if:</p> <ul style="list-style-type: none"> • either the condition expression result is <i>true</i>; or • the condition expression is left empty. <p>For details see Condition Expressions (Approval Workflow).</p> |
| Skip | <p>If set to Yes, the condition is not checked (not used) and so the approver will always be in the workflow.</p> <p>If set to No, the condition is checked.</p> |

Note:

You can combine the condition of values on a header and line item, line item and line item of the same level within one condition row. But there is currently a limitation: when combining a line item with another line item of a different level (folder), you have to have two separate rows.

Further reading:

- [Condition Expressions \(Approval Workflow\)](#)
- [Values for Condition Expressions \(Approval Workflow\)](#)
- [Lookup Tables Support in Conditions \(Approval Workflow\)](#)
- [Add More Approvers to One Step \(Approval Workflow\)](#)

Condition Expressions (Approval Workflow)

The [Price Parameter ApprovalConditions](#) is used to set up conditions under which the approval step will appear in the approval workflow.

Price Parameter Values: Approval conditions at each step

+ Add Record ...   

| <input type="checkbox"/> | Workflow Type | Step ID | Condition ID | Condition Description | Condition | Skip |
|--------------------------|--------------------------------|------------------------------|--------------------------------|--------------------------------|--|-----------------------------------|
| | Search... <input type="text"/> | Sear... <input type="text"/> | Search... <input type="text"/> | Search... <input type="text"/> | Search... <input type="text"/> | Select Value <input type="text"/> |
| <input type="checkbox"/> | PriceGridItem | 1 | 1 | Local Price | (LocalPrice > 1.5) AND ((0.8 + 0.7) <= Adjustment) | No |
| <input type="checkbox"/> | PriceList | 1 | 1 | No condition | | No |
| <input type="checkbox"/> | Quote | 1 | 1 | Total Revenue | header.TotalRevenue > 100 && quoteType == 'CPQ' | No |
| <input type="checkbox"/> | Quote | 2 | 1 | MarginPct | header.TotalRevenue > 500 AND line.MarginPctOutput <= 0.55 | No |
| <input type="checkbox"/> | Quote | 3 | 1 | Quantity | line.folder.TotalQuantity > 100 AND label == "MB" | No |
| <input type="checkbox"/> | RebateRecord | 1 | 1 | Check Rebate | attribute2 >= 1000 | No |
| <input type="checkbox"/> | Rebates | 1 | 1 | Check Target Input | line.Target < 20000 | No |

Decisions when to add a step:

- If there is no condition for a step, or all step's conditions have the field **Skip** set to *Yes*, the step will always be added.
- If there are more conditions for the same step, then all of them must be *true*, so that the step is added to the workflow.

This Boolean expression can consist of:

- Names of fields
- Names of inputs and outputs/results
- Operators - described at [Expressions Language \(Formula Evaluator Library\)](#)
- Numbers - if you need a decimal number, use the decimal point to separate the whole part from the fractional part
- Boolean value - either *true* or *false*
- Text - must be in enclosed in either double quotes or single quotes

The rule for setting a condition for a header, line items or folder:

- header.ElementName + condition
- line.ElementName + condition
- line.folder.ElementName + condition AND label == "folder name" (if any)

Note:

- This rule is helpful when you have the same element names on both header and line item level - using a rule you can distinguish them.

- `header.ElementName` and `line.ElementName` are only applied for the input or output. For example, you want to set a condition for Product Id (MB-0035), you have to set like this: `sku == "MB-0035"`.

Examples:

- An LPG does not contain a line item, so you can set `ElementName + condition` (suggested) or `header.ElementName + condition`.
- A Price List contains a line item, so you can set `ElementName + condition` (suggested) or `line.ElementName + condition`.

Which fields, inputs and outputs will be available in the expression depends on what you are building the approval workflow for.

If there are multiple condition lines for one step, the conditions are combined using AND operator.

i You can combine the condition of values on a header and line item, line item and line item of the same level within one condition row. But there is currently a limitation when combining a line item with another line item of a different level (folder), then you have to have two separate rows.

Process of Conditions Evaluation

When the system needs to build the approval workflow steps for certain document:

1. It will execute the appropriate workflow logic based on the type of document.
2. That workflow logic provides a name of the `Workflow Type`, which will be used to select/filter only the relevant steps and their conditions.
3. All active conditions (i.e. their **Skip** field is not `Yes`) defined for that `Workflow Type` will be evaluated against the document and also against every its line items (when it has line items).
4. Approval steps will be added to the approval workflow based on results of the conditions.

Fields Available in Expression

The condition can read values of the fields of the document simply by using the name of the field, e.g. `sku`, `label`, `targetDate`, `unitOfMeasure`, `quoteType`, `attribute2`, etc.

When the system evaluates the condition for a line item in a document which also has a header, then the fields from the line item and header are both available. For example, in a condition for a quote line item, you can refer to `targetDate` (from quote header) and also to `sku` (from quote line).

In rare cases where the field exists in both - item and header - the line field value will be used.

Values of Inputs and Outputs Available in Expression

Values provided by the user in the input fields and outputs (result values calculated by the logic and stored on the document) are available via their name (not labels), e.g. `TotalRevenue`, `LocalPrice`.

If you have inputs/outputs with the same names on the header and line item levels, you need to distinguish and use *header* and *line* prefix, e.g. `header.Quantity`, `line.Quantity`.

If you have an input and also output with the same name, you need to distinguish which one you want, e.g. `line.inputs.Quantity`, `line.outputs.Quantity`.

If an input field name is defined with spaces in the name, then you can put its name into quotes, e.g. `line."Discount %"`. Note that you can do it only when using the prefix *line* or *header*.

Examples per Document Type

Here are some examples of what fields/input/outputs you can refer to in the condition expression.

Live Price Grid Item or Price List Fields/Inputs/Outputs

Please remember that Live Price Grid has approval workflow per line item, whereas Price List has a single approval workflow for the whole price list.

| Field/output available in condition expression | Meaning of the value | Value comes from |
|--|---------------------------------------|---------------------------|
| sku | Product ID | Price list/ LPG Line Item |
| label | Product Name | Price list/ LPG Line Item |
| unitOfMeasure | UOM | Price list/ LPG Line Item |
| currency | Product Currency | Price list/ LPG Line Item |
| resultPrice | Result Price | Price list/ LPG Line Item |
| manualResultPrice | Manual Override | Price list/ LPG Line Item |
| <ResultName> | Calculation result named <ResultName> | Price list/ LPG Line Item |
| line.<ResultName> | Calculation result named <ResultName> | Price list line item |
| header.<ResultName> | Calculation result named <ResultName> | LPG line item |

Quote Fields/Inputs/Outputs

| Field/output available in condition expression | Meaning of the value | Value comes from |
|--|---------------------------------------|------------------|
| targetDate | Effective Date | Header |
| expiryDate | Expiry Date | Header |
| header.<InputName> | input field named <InputName> | Header |
| header.<ResultName> | calculation result named <ResultName> | Header |
| label | Product Name | Line/Folder |
| sku | Product Id | Line |
| quoteType | Quote Type | Line |
| line.<InputName> | input field | Line |
| line.<ResultName> | calculation result | Line |
| line.folder.<InputName> | input field named <InputName> | Folder |
| line.folder.<ResultName> | calculation result named <ResultName> | Folder |

Rebate Agreement Fields/Inputs/Outputs

| Field/output available in condition expression | Meaning of the value | Value comes from |
|--|---------------------------------------|------------------|
| startDate | Start Date | Header |
| endDate | End Date | Header |
| payoutDate | Payout Date | Header |
| rebateType | Rebate Type | Line |
| line.<InputName> | input field named <InputName> | Line |
| line.<ResultName> | calculation result named <ResultName> | Line |

Values for Condition Expressions (Approval Workflow)

When Approval Workflow is used to build approval steps within other accelerators, you need to know which values you can use in the condition expressions. This page lists the names of values (possibly called also results/outputs/elements) you can refer to when building approval workflow conditions for that accelerator.

Based on [Quick Technical Overview \(Approval Workflow\)](#), this section provides the element names used in Accelerators. Using these values you can set your own condition in the configuration.

- [Accelerate Price Setting Package \(LPG and Price List\)](#)
- [Accelerate CPQ Package](#)
- [Accelerate Rebate Management Package](#)
- [Accelerate Sales Compensation Package](#)

Accelerate Price Setting Package (LPG and Price List)

| Label | Element Name |
|---------------------------|-------------------|
| Product Id | sku |
| Product Name | label |
| UOM | unitOfMeasure |
| Product Currency | currency |
| Result Price | resultPrice |
| Manual Override | manualResultPrice |
| Actual Price Lookup | ActualPriceLookup |
| Dependency Level Currency | BaseCurrency |
| Advanced Costs | <not supported> |
| Cost | Cost |

| | |
|---------------------------|---------------------|
| Competition Data | <not supported> |
| Relevant Competition Data | <not supported> |
| Discount Group | DiscountGroup |
| Discount | Discount |
| Volume Discount | VolumeDiscount |
| Minimum Margin Price | MinimumMarginPrice |
| Stock | Stock |
| Days of Cover | StockCoverDays |
| Prices | Prices |
| Price Selector | PriceSelector |
| Override Price | ManualPrice |
| Override Reason | ManualPriceReason |
| Exceptions | Exceptions |
| Final List Price | FinalListPrice |
| Net Price | NetPrice |
| Final Price | FinalPrice |
| Price Decision | PriceDecision |
| Margin | Margin |
| Minimum Margin | MinimumMargin |
| Sales Volume YTD | SalesVolumeYTD |
| Turnover YTD | TurnoverYTD |
| Last Year Sales Volume | LastYearSalesVolume |
| Last Year Turnover | LastYearTurnover |
| Sales Volume Forecast | SalesVolumeForecast |
| Turnover Forecast | TurnoverForecast |
| Price Change Effect | PriceChangeEffect |

| Header Level Label | Element Name |
|------------------------------|---|
| Effective Date | targetDate |
| Expiry Date | expiryDate |
| Customer | header.Customer |
| Discount Type | header.DiscountConfigurator."Discount Type" |
| Pricing Detail | |
| Currency | Currency |
| Sales Overview | <not supported> |
| Additional Discount | header.AdditionalDiscount |
| Revenue | header.Revenue |
| Cost | header.Cost |
| Net Price | header.NetPrice |
| Margin % | header.MarginPct |
| Margin | header.Margin |
| Quantity | header.Quantity |
| Total Deal Score % | header.DealScorePercent |
| Total Deal Score | header.DealScore |
| Customer History | |
| Customer Revenue | header.customerRevHistory |
| Customer Quantity | header.customerQuantityHistory |
| Customer Margin | header.customerMarginHistory |
| Customer Margin % | header.customerMarginPctHistory |
| Line Item Level Label | Element Name |
| Product Name | label |
| Product Id | sku |
| Quantity in Default Inputs | line.DefaultInputConfigurator.Quantity |
| Price in Default Inputs | line.DefaultInputConfigurator.Price |
| Discount % in Default Inputs | line.DefaultInputConfigurator."Discount %" |

| Pricing Detail | |
|-----------------------------|-------------------------------------|
| Quantity | line.Quantity |
| List Price | line.ListPrice |
| Discount Amount | line.DiscountAmount |
| Discount % | line.DiscountPct |
| Invoice Price | line.InvoicePrice |
| Rebate | line.Rebate |
| Cash Discount % | line.CashDiscount |
| Net Price | line.NetPrice |
| Cost | line.Cost |
| Margin | line.Margin |
| Margin % | line.MarginPct |
| Revenue | line.Revenue |
| Product History | |
| Historical Period | line.HistoricalPeriod |
| Product Revenue | line.ProductRevenueHistory |
| Product Quantity | line.ProductQuantityHistory |
| Product Margin | line.ProductMarginHistory |
| Product Margin % | line.ProductMarginPctHistory |
| Product Weighted Avg. Price | line.ProductWeightedAvgPriceHistory |
| Avg. Revenue Per Invoice | line.AvgRevenuePerInvoiceHistory |
| Avg. Quantity Per Invoice | line.AvgQuantityPerInvoiceHistory |
| Avg. Margin % Per Invoice | line.AvgMarginPctPerInvoiceHistory |
| Previous Price | |
| Unit Price | line.PreviousUnitPrice |
| Margin % | line.PreviousMarginPct |
| Date | line.PreviousDate |
| Competitor | |

| | |
|---------------------------|----------------------------|
| Competitor Median | line.CompetitorMedian |
| Top 5 Competitors | <not supported> |
| Lowest Competitor Price | line.LowestCompetitorPrice |
| Price Guidance | |
| Target Price | line.TargetPrice |
| Floor Price | line.FloorPrice |
| Stretch Price | line.StretchPrice |
| Deal Score % | line.DealScorePercent |
| Deal Score | line.DealScore |
| Folder Level Label | Element Name |
| Revenue | line.folder.Revenue |
| Cost | line.folder.Cost |
| Net Price | line.folder.NetPrice |
| Margin % | line.folder.MarginPct |
| Margin | line.folder.Margin |
| Quantity | line.folder.Quantity |

Accelerate Rebate Management Package

| Label | Element Name |
|------------------------------|---------------------|
| Start Date | startDate |
| End Date | endDate |
| Payout Date | payoutDate |
| Line Item Level_Label | Element Name |
| Rebate Type | rebateType |
| Input | |
| Target | line.Target |
| Rebate | line.Rebate |
| Rebate % | line."Rebate %" |

| | |
|----------------------------|--|
| Increment | line.Increment |
| Target Quantity | line."Target Quantity" |
| Rebate Per Unit | line."Rebate Per Unit" |
| Payment Period | line."Payment Period" |
| Rebate Detail | |
| Baseline Value Field Label | line.BaselineValueFieldName |
| Previous Baseline Value | line.PreviousBaselineValue |
| Current Baseline Value | line.CurrentBaselineValue |
| Previous Rebate | line.PreviousRebate |
| Current Rebate | line.CurrentRebate |
| Forecast | |
| Forecast Baseline Value | line.ForecastBaselineValue |
| Forecast Rebate | line.ForecastRebate |
| Rebate Type Info | |
| Target For | line.Display_RebateTypeInfo_TargetFor |
| Rebate Formulas | line.Display_RebateTypeInfo_RebateFormulas |
| Payment Period | line.Display_RebateTypeInfo_PaymentPeriod |
| Pay To | line.Display_RebateTypeInfo_PayTo |
| Customer(s) | <not supported> |
| Product(s) | <not supported> |

Accelerate Sales Compensation Package

See [Workflow Variables \(SC\)](#).

Lookup Tables Support in Conditions (Approval Workflow)

You can define data obtained from P, C, PX, CX, or PP and use them in an approval workflow condition expression.

For example, we have a quote condition that starts an approval process: `line.MarginPct < noWayMargin`. In the condition, `line.MarginPct` is a margin percentage per line item in the Quote and `noWayMargin` is the expected margin percentage for each line item (the quote should not go lower than this value) and its value is fetched from the PX, PP, etc. table. The previous version looked like `line.MarginPct < 0.1`.

In this improved version, we provide two Price Parameter tables besides `ApprovalWorkflowSetup`,

Approvers, and ApprovalConditions. To provide better guidance, we created an example PP table "Test Expected Margin Quote":

| Price Parameter Values : Test Quote Approval For Expected Margin [3] | | | | | |
|--|-------------|------------|-------------------|-----------------|--------------------------|
| <input type="checkbox"/> | Customer Id | Product Id | Very Low Margin % | No Way Margin % | Below Preferred Margin % |
| <input type="checkbox"/> | CD-00001 | BV-0007 | 10.00 % | 20.00 % | 34.00 % |
| <input type="checkbox"/> | CD-00001 | MB-0002 | 12.00 % | 3.00 % | 45.00 % |
| <input type="checkbox"/> | CD-00001 | MB-0001 | 67.00 % | 68.00 % | 71.00 % |

Set up Price Parameter Tables

WFExpressionVariableConfiguration

| Column Name | Description |
|----------------|---|
| Variable Name | Variable that can be used in the condition expression. You can define it in your own way but the input should look like noWayMarign, expectedMargin, etc. |
| Workflow Type | Name of the workflow type (Quote, Contract, PriceList, PriceGridItem, etc.) where the variable is used. |
| Lookup Table | Defines a lookup table seeking for the value. |
| Lookup Type | Defines the lookup type. Supported types: P, C, PX, CX, PP. |
| Variable Value | Defines an attribute of the lookup table where the variable value is stored. |
| Variable Type | Defines the variable data type for auto type casting. Supported values: Number (default), String. |
| Skip | "Yes" - Allows to skip the variable name "No" - Keeps the variable name |

Example:

| Price Parameter Values : Additional Variable [24] | | | | | | | |
|---|------------------------|---------------|-------------------------|-------------|--------------------------|---------------|------|
| <input type="checkbox"/> | Variable Name | Workflow Type | Lookup Table | Lookup Type | Variable Value | Variable Type | Skip |
| <input type="checkbox"/> | veryLowMargin | Quote | TestExpectedMarginQuote | PP | attribute1 | Number | |
| <input type="checkbox"/> | noWayMargin | Quote | TestExpectedMarginQuote | PP | attribute2 | Number | |
| <input type="checkbox"/> | belowPreferredMargin | Quote | TestExpectedMarginQuote | PP | Below Preferred Margin % | Number | Yes |
| <input type="checkbox"/> | Discount | Quote | CPQExceptionDiscount | PP | attribute1 | | |
| <input type="checkbox"/> | isPercentage | Quote | CPQExceptionDiscount | PP | attribute2 | String | |
| <input type="checkbox"/> | folderBeverages | Quote | TestFolderQuoteApproval | PP | attribute1 | Number | |
| <input type="checkbox"/> | folderSpecificProducts | Quote | TestFolderQuoteApproval | PP | attribute2 | | |
| <input type="checkbox"/> | BusinessUnit | Quote | Product | P | attribute1 | String | |
| <input type="checkbox"/> | floorPrice | Quote | PriceGuidance | PX | attribute2 | Number | |
| <input type="checkbox"/> | currentRebate | Rebates | TestRebateApproval | PP | attribute1 | Number | |
| <input type="checkbox"/> | rebateTypePP | Rebates | TestRebateApproval | PP | attribute2 | String | |
| <input type="checkbox"/> | rebateAmount | Rebates | TestRebateApproval | PP | attribute3 | Number | |
| <input type="checkbox"/> | region | Quote | Customer | C | attribute4 | String | |
| <input type="checkbox"/> | rebateRecord | RebateRecord | TestRebateApproval | PP | attribute4 | Number | |
| <input type="checkbox"/> | productLifeCycle | Quote | Product | P | attribute4 | String | |
| <input type="checkbox"/> | country | Quote | Customer | C | attribute5 | String | |
| <input type="checkbox"/> | priceDecision | PriceList | TestPriceListApproval | PP | attribute1 | String | |
| <input type="checkbox"/> | finalPrice | PriceList | TestPriceListApproval | PP | attribute2 | Number | |
| <input type="checkbox"/> | targetPrice | Quote | PriceGuidance | PX | attribute1 | Number | |
| <input type="checkbox"/> | cost | Quote | CostData | PX | attribute2 | Number | |
| <input type="checkbox"/> | cashDiscount | Quote | CustomerCashDiscount | CX | attribute1 | Number | |
| <input type="checkbox"/> | cashDiscountPct | Quote | TestCashDiscount | CX | attribute1 | Number | |
| <input type="checkbox"/> | priceDecision | PriceGridItem | TestPriceListApproval | PP | attribute1 | String | |
| <input type="checkbox"/> | finalPrice | PriceGridItem | TestPriceListApproval | PP | attribute2 | Number | |

WFLookupFilterConfiguration

| Column Name | Description |
|----------------|---|
| Variable Name | Variable name from the WFExpressionVariableConfiguration PP table to define the lookup key. |
| Lookup Table | Defines a value used to look up for each key. |
| Lookup Key | Defines a key. For example, key1, key2 for CustomerId, ProductId in the Test Expected Margin Quote PP shown above. |
| Lookup Value | Defines a value where the key will get data. For example, key1 will get data from customerId in Customer Master, key2 is from sku in Product Master. Note: For a Rebate Agreement with the input type Product Group / Customer Group, you must define "customerGroup" or "productGroup" in this column. |
| Value Provider | Defines a value provider if the lookup value can not be directly accessed from the approvable. There are some predefined values: |

- **customer** - Gets a value in a specific attribute for customerId (if the lookup value is not obtained directly from customerId, it is from another attribute, e.g. Country in Customer Master).
- **product** - Gets a value in a specific attribute from sku (if the lookup value is not obtained directly from sku, it is from another attribute, e.g. Business Unit in Product Master).
- **customerGroup** - Gets customerFieldValue from the customer group input.
- **productGroup** - Gets productFieldValue from the product group input.

If you are getting the values directly from customerId or sku, you do not need to configure this column.

Example:

| <input type="checkbox"/> | Variable Name | Lookup Table | Lookup Key | Lookup Value | Value Provider |
|--------------------------|------------------------|-------------------------|------------|---------------|----------------|
| <input type="checkbox"/> | cost | CostData | sku | sku | |
| <input type="checkbox"/> | isPercentage | CPQExceptionDiscount | key1 | customerId | |
| <input type="checkbox"/> | isPercentage | CPQExceptionDiscount | key2 | sku | |
| <input type="checkbox"/> | Discount | CPQExceptionDiscount | key1 | customerId | |
| <input type="checkbox"/> | Discount | CPQExceptionDiscount | key2 | sku | |
| <input type="checkbox"/> | region | Customer | customerId | customerId | |
| <input type="checkbox"/> | country | Customer | customerId | customerId | |
| <input type="checkbox"/> | cashDiscount | CustomerCashDiscount | customerId | customerId | |
| <input type="checkbox"/> | targetPrice | PriceGuidance | sku | sku | |
| <input type="checkbox"/> | productLifeCycle | Product | sku | sku | |
| <input type="checkbox"/> | BusinessUnit | Product | sku | sku | |
| <input type="checkbox"/> | floorPrice | ProductPriceGuidance | sku | sku | |
| <input type="checkbox"/> | cashDiscountPct | TestCashDiscount | customerId | customerId | |
| <input type="checkbox"/> | veryLowMargin | TestExpectedMarginQuote | key1 | customerId | |
| <input type="checkbox"/> | veryLowMargin | TestExpectedMarginQuote | key2 | sku | |
| <input type="checkbox"/> | noWayMargin | TestExpectedMarginQuote | key1 | customerId | |
| <input type="checkbox"/> | noWayMargin | TestExpectedMarginQuote | key2 | sku | |
| <input type="checkbox"/> | belowPreferredMargin | TestExpectedMarginQuote | key1 | customerId | |
| <input type="checkbox"/> | belowPreferredMargin | TestExpectedMarginQuote | key2 | sku | |
| <input type="checkbox"/> | folderSpecificProducts | TestFolderQuoteApproval | name | attribute2 | customer |
| <input type="checkbox"/> | folderBeverages | TestFolderQuoteApproval | name | attribute2 | customer |
| <input type="checkbox"/> | priceDecision | TestPriceListApproval | name | sku | |
| <input type="checkbox"/> | finalPrice | TestPriceListApproval | name | sku | |
| <input type="checkbox"/> | rebateTypePP | TestRebateApproval | key1 | customerGroup | customerGroup |
| <input type="checkbox"/> | rebateTypePP | TestRebateApproval | key2 | productGroup | productGroup |
| <input type="checkbox"/> | rebateRecord | TestRebateApproval | key1 | customerGroup | customerGroup |
| <input type="checkbox"/> | rebateRecord | TestRebateApproval | key2 | productGroup | productGroup |
| <input type="checkbox"/> | rebateAmount | TestRebateApproval | key1 | customerGroup | customerGroup |
| <input type="checkbox"/> | rebateAmount | TestRebateApproval | key2 | productGroup | productGroup |
| <input type="checkbox"/> | currentRebate | TestRebateApproval | key1 | customerGroup | customerGroup |
| <input type="checkbox"/> | currentRebate | TestRebateApproval | key2 | productGroup | productGroup |

Example for approval condition:

| Workflow... | Step ID | Condition ID | Condition Description | Condition | Skip |
|---------------|---------|--------------|--|--|------|
| PriceGridItem | 1 | 1 | Check LPG | "header.FinalPrice" < finalPrice | |
| PriceList | 1 | 1 | Check Price Decision and Final Price | FinalPrice < finalPrice AND PriceDecision == priceDecision | |
| Quote | 1 | 1 | Margin % less than Very Low Margin % | IF(isPercentage == "%", "line.DiscountPct" > Discount / 100, "line.DiscountAmount" > Discount) | |
| Quote | 2 | 1 | Margin % less than No Way Margin % | "line.MarginPct" != null AND "line.MarginPct" < noWayMargin | |
| Quote | 3 | 1 | Margin % less than Below Preferred Margin % | "line.MarginPct" < belowPreferredMargin | |
| Quote | 4 | 1 | Some folders are less than expected Margin % | "line.folder.MarginPct" < folderBeverages AND label == "Beverages" | |
| Quote | 5 | 1 | Check Business Unit | BusinessUnit == "Food" AND productLifeCycle == "Decline" | |
| Quote | 6 | 1 | Check Net Price with Floor Price | "line.NetPrice" > IF(targetPrice == null, 9999, targetPrice) | |
| Quote | 7 | 1 | Check Region of Customer | region == "Europe" AND country == "Italy" | |
| Quote | 4 | 2 | Some folders are less than expected Margin % | "line.folder.MarginPct" < folderSpecificProducts AND label == "Specific Products" | |
| Quote | 8 | 1 | Check different PX | "line.Cost" < cost | |
| Quote | 9 | 1 | Check CX | "line.MarginPct" > cashDiscount AND "line.CashDiscount" < cashDiscountPct | |
| RebateRecord | 1 | 1 | Check Rebate Record | attribute2 < rebateRecord | |
| Rebates | 1 | 1 | Check Current Rebate | "line.CurrentRebate" < 100 | |
| Rebates | 2 | 1 | Check Rebate amount, Rebate Type | rebateType == rebateTypePP AND "line.Rebate" < rebateAmount | |
| Rebates | 3 | 1 | Check Rebate value map | Target > 1 | |
| Rebates | 4 | 1 | Check other rebate conditions | line."Payment Period" == "Semi-Annually" | |

Add More Approvers to One Step (Approval Workflow)

To add more approvers for the same step, add more records to the ApprovalWorkflowSetup table, with the same **WorkflowType** and **Step ID**, but with different values in **Approver ID** field.

Price Parameter Values: Approvers at each step



| Workflow Type | Step ID | Approver ID | Approver Type | Approver Name | Skip |
|--------------------------------|------------------------------|--------------------------------|------------------------------------|--------------------------------|-----------------------------------|
| Search... <input type="text"/> | Sear... <input type="text"/> | Search... <input type="text"/> | Select Val... <input type="text"/> | Search... <input type="text"/> | Select Value <input type="text"/> |
| PriceGridItem | 1 | 1 | USER GROUP | Admin | No |
| PriceList | 1 | 1 | USER GROUP | SalesManager | No |
| Quote | 1 | 1 | USER | joe.smith | No |
| Quote | 1 | 2 | USER | john.price | No |
| Quote | 2 | 1 | USER GROUP | SalesManager | No |
| RebateRecord | 1 | 1 | BUSINESS ROLE | BusinessRole01 | No |
| Rebates | 1 | 1 | USER | vinh | No |

Limitations

For a single step you can only combine approvers with the same **Approver Type**. For example, do not combine the approver record with **Approver Type** USER and USER GROUP.

Workflow Logics (Approval Workflow)

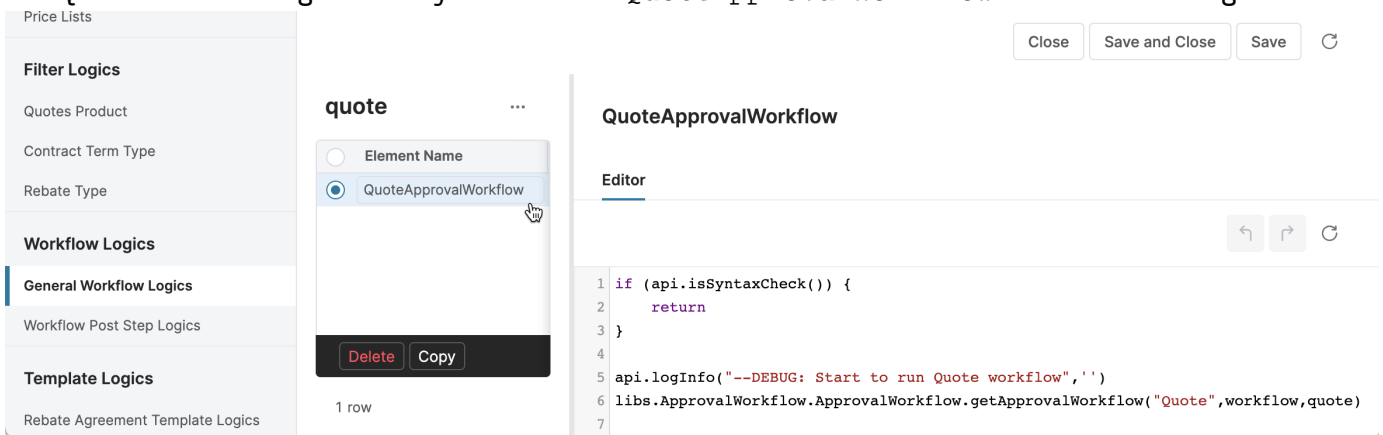
During installation of this accelerator as the Approval Workflow Package, you were asked if you want to install [sample workflow logics](#). If you installed them, then you already have a workflow logic ready to be used.

To review the sample logics, in Pricefx Unity UI, navigate to **Configuration > Logics > Workflow Logics > General Workflow Logics**.

Those logics are used by the system automatically, thanks to their Workflow Type and special name of the logic. Even for Quotes you do not need to register them in the Quote Type, unless you explicitly want to.

Example:

The Quote workflow logic has only one element `QuoteApprovalWorkflow` with the following code:



The screenshot shows the Pricefx Unity UI interface for configuring workflow logics. On the left, a sidebar menu is visible with categories like 'Filter Logics', 'Workflow Logics', and 'Template Logics'. Under 'Workflow Logics', 'General Workflow Logics' is selected, showing a table with one row for 'quote'. The main area displays the configuration for the 'quote' logic, including an 'Editor' window with the following code:

```
1 if (api.isSyntaxCheck()) {
2     return
3 }
4
5 api.logInfo("--DEBUG: Start to run Quote workflow",'')
6 libs.ApprovalWorkflow.ApprovalWorkflow.getApprovalWorkflow("Quote", workflow, quote)
7
```

Create Your Own Logic

If you installed the Approval Workflow Package without the sample logics, or you installed only the Approval Workflow Library, then you need to create the workflow logic on your own.

Steps:

1. Create a new General (default type) Workflow Logic.
2. Add one element with the code which will call the Accelerator library to build the workflow steps, based on the configuration done in the price parameters.

```
if (api.isSyntaxCheck()) return

libs.ApprovalWorkflow.ApprovalWorkflow
    .getApprovalWorkflow("Quote", workflow, quote)
```

- The function is defined as `def getApprovalWorkflow(String workflowType, Object workflow, Object approvableObject)` where the `workflowType` string must match what you use in the configuration price parameters; bindings variables `workflow` and `quote` are available in a workflow logic by default.
3. In case of a workflow for Quotes, if you did not use the default name of the logic *quote*, you need to register the workflow logic in the Quote Type.

Notes: If you change the name, check the workflow logic again because it will use the default setup after each deployment.

4. If you want to (de)activate a workflow type, set it in the Status column.

Formula Evaluator Library (Approval Workflow)

This package uses [Expressions Language \(Formula Evaluator Library\)](#) to resolve approval conditions.

Troubleshooting (Approval Workflow)

Price List

When you click the **View Workflow** button and you get an error `Workflow Error: Workflow: Please check Approver Name in Price Parameter table!` then review the **Approvers** price parameter. The field **ApproverName** is probably empty.

Quote

When you navigate to the **Workflow** tab or click the **Submit** button and you get an error `Workflow Error: Workflow: Please check Approver Name in Price Parameter table!` then review the **Approvers** price parameter. The field **ApproverName** is probably empty.

Release Notes (Approval Workflow)

- [Approval Workflow Package 1.2.0](#)
- [Approval Workflow Package 1.1.2](#)
- [Approval Workflow Package 1.1.0](#)

Approval Workflow Package 1.2.0

Upgrade Notes

- The PP names are adjusted. So, the workflow configurations need to be migrated manually to the new PPs.
- The step order column is not used in the new PP and the step Id will be used instead. So, you need to add the number to the step Id to maintain the step order, eg. 000001_stepID1, 000002_stepID2, etc.
- The condition with the level prefix and the quote needs to be changed by removing the quote, e.g. `"line.Quantity" > 0` -> `line.Quantity > 0`

Stories

[PFPCS-2674](#) Support setting condition for configuration and formulaParameter on PL

[PFPCS-4571](#) Set default preferences for PP tables

- PFPCS-4892 Remove StepId from package configuration
- PFPCS-4893 Adjust PP names in Approval WF Package
- PFPCS-5520 Add missing Confluence URL to Approval Workflow Library package
- PFPCS-6325 Evaluator configurable configurations

Improvements

- PFPCS-3417 Add FormulaEvaluator as package dependency
- PFPCS-4343 Remove default values from Approval WF Package PPs
- PFPCS-4692 Add level prefix for all fields in approvable object
- PFPCS-6241 Condition evaluation improving
- PFPCS-6247 Approvable object enhancement process improving
- PFPCS-5108 Add Sales Compensation example logic

Bugs

- PFPCS-2672 WF enhancement - Handle case if it's not input, output
- PFPCS-4678 Approval Workflow - YES, NO is displayed in sample data
- PFPCS-4679 WF - Incorrect behavior when using USER and USERGROUP in 1 step
- PFPCS-4684 Missing header binding map on line item
- PFPCS-4950 DealPlan sample logic not offered in the deployment step
- PFPCS-5347 WF condition rebateType == rebateTypePP does not work
- PFPCS-5359 WF only works with 1 condition
- PFPCS-5361 There is no workflow appeared after clicking Submit button
- PFPCS-5405 WF: agreement record does not work
- PFPCS-5516 WF: Step order is not shown properly if condition contains header
- PFPCS-6186 "Please check value of Approver Name in Price Parameter table." is displayed when opening Workflow tab
- PFPCS-6207 Using double quote in expression does not work anymore in current version
- PFPCS-6216 Adjustment workflow step is not displayed after submission
- PFPCS-6321 SC workflow error when trying to click on Workflow tab
- PFPCS-6721 Using 'sku' as lineId in Matrix Price List is not sufficient.

Approval Workflow Package 1.1.2

Bugs

- PFPCS-4109 OOM when start approval for approvable object contains many line items

Improvements

[PFPCS-3640](#) Support LargeDealOptimizer Plan Types

Tasks

[PFPCS-3876](#) Add documentation link to Approval Workflow package description

Approval Workflow Package 1.1.0

- [Stories](#)
- [Improvements](#)
- [Sub-tasks](#)
- [Bugs](#)
- [Tasks](#)

Stories

- [\[PFPCS-627\]](#) - Add support for lookup tables
 - **New Price Parameters introduced**
 - `WFExpressionVariableConfiguration`
 - `WFLookupFilterConfiguration`
 - For more information see [Accelerate Approval Workflow Package](#)
- [\[PFPCS-831\]](#) - Approval workflow configuration POC
- [\[PFPCS-1668\]](#) - Use name/label in configuration instead of technical name

Improvements

- [\[PFPCS-1329\]](#) - WF Enhancement_Dropbox for some pre-defined columns in PP table
- [\[PFPCS-1559\]](#) - WF Enhancement_Check table if it exist or not in PX, PP,..
- [\[PFPCS-2353\]](#) - Enhance Performance for Workflow logic in PL

Sub-tasks

- [\[PFPCS-125\]](#) - Support when value return map
- [\[PFPCS-742\]](#) - Additional binding PP configuration definition
- [\[PFPCS-743\]](#) - Additional binding parser
- [\[PFPCS-744\]](#) - Approval expression enhance with additional binding
- [\[PFPCS-745\]](#) - Create Predefined PP for additional binding
- [\[PFPCS-1258\]](#) - ConfigManager - getting configurations
- [\[PFPCS-1259\]](#) - Refactor current implementation using ConfigManager
- [\[PFPCS-1262\]](#) - Documentation update for lookup tables supported
- [\[PFPCS-1311\]](#) - Filter notEqual not work on null/ empty column
- [\[PFPCS-1552\]](#) - Multiple binding support
- [\[PFPCS-1676\]](#) - Additional Variable skip option
- [\[PFPCS-1677\]](#) - Verify Lookup table before query

Bugs

- [\[PFPCS-1012\]](#) - FormulaEvaluator: Sonar Error - The nested block depth is 6
- [\[PFPCS-1013\]](#) - FormulaEvaluator: Sonar Error - println should be replaced with something more robust
- [\[PFPCS-1014\]](#) - FormulaEvaluator: Sonar CodeSmell - The catch block is empty
- [\[PFPCS-1264\]](#) - Approval workflow enhancement: invalid result when lookup multiple variables value

- [PFPCS-1265] - Additional variable configuration is not validated
- [PFPCS-1266] - Approval workflow enhancement: variable value can not be mapped if using value provider
- [PFPCS-1308] - Error @152 for Workflow Utils_Approval enhancement
- [PFPCS-1313] - WF Enhancement_Show approval workflow if output is empty
- [PFPCS-1339] - WF enhancement_Skip checking all lines when throw exception
- [PFPCS-1350] - WF enhancement_Not work if lookup from Product master
- [PFPCS-1381] - WF Enhancement_Not work if lookup from PX, CX
- [PFPCS-1453] - Approval Workflow not work LPG, PL
- [PFPCS-1550] - Error parse number
- [PFPCS-1741] - WF enhancement - Not work when using condition "line.folder"
- [PFPCS-1742] - Run condition incorrectly when compare "line.CashDiscount" > cashDiscountPct
- [PFPCS-1747] - Timed out for PriceListApprovalWorkflow with large number of items (1000+)
- [PFPCS-1748] - Approval workflow not work for some PLs
- [PFPCS-2080] - Conflict between master and dev
- [PFPCS-2238] - Workflow error is displayed with No such property in LPG and not work in PL also
- [PFPCS-2338] - Approval Workflow does not work when VolumeBreakdownMagicLogic is used in PL
- [PFPCS-2570] - WF Enhancement_Can not work if using label in WFExpressionVariableConfiguration PP
- [PFPCS-2629] - WF Enhancement - LPG does not work both in Simple and Matrix
- [PFPCS-2632] - WF Enhancement - Rebate does not work
- [PFPCS-2666] - WF enhancement - Can't set the condition line."Payment Period"
- [PFPCS-2669] - WF does not work if setting the condition quoteType == "Value Estimator"
- [PFPCS-2992] - WF enhancement - Issue with Operator
- [PFPCS-3011] - Approval Workflow_Missing Formula Evaluator dependency

Tasks

- [PFPCS-3002] - WF enhancement_Set dropdown and preference from other tables