



Accelerate Sales Compensation

Version 1.7.1

March 2023

Accelerate Sales Compensation

The Sales Compensation Accelerator provides a solution for fast implementation of the company's sales compensation plan. Its purpose is to maintain, control and evaluate execution with Pricefx application. The package includes:

- Common compensation types
- Approval workflow for compensation agreements, agreement records and payout values
- Output templates for documents like Sales Compensation Agreement, configuration overview and progress reports
- Dashboards with information about progress towards the plan
- Dedicated views for compensation agreements and agreement records

> iDPN - circuit breaker - iDPN N - 1P + N - 13A - ...

Input Parameters **Calculations**

Lowest Competi...	120.60	€
-------------------	--------	---

▼ Sales Compensation

Label	Calculation Result	Message
Sales Represent...	admin	
Quote ID	P-879	
Expected Turnov...	57.48	€
Compensation %	0.00%	
Currency	EUR	
Compensation	0.00	€

▼ Taxes

Key Capabilities

- **Define** and **create** agreements based on different compensation types.
- Give sales rep **visibility** on their potential compensation **while creating a quote**.
- **Track** and **analyze** your sales team's compensation's progress over time.
- **Approve** agreements and connect to a third-party solution to **payout** sales reps.

[SC] Stepped Percent

Input Parameters **Calculations**

Label	Calculation Result	Message
No data		

▼ Compensation Detail

Label	Calculation Result	Message
Baseline Value FI...	Invoice Price	
Previous Baselin...	310.20	
Current Baseline...	10,509.47	
Previous Compe...	6.22	
Current Compen...	1,137.4	

▼ Compensation Type Info

Label	Calculation Result	Message
Target For	Payment Period	
Compensation F...	Non-Cumulative	
Payment Period	Quarterly	
Customer(s)		
Product(s)		

pricefx Dashboards

Dashboard Settings

Select Dashboard
Seller Overview

DATA FILTER

Seller*
Seller 002 (SC-002)

Compensation Plan(s)
CO-932 | New Compensation Plan X

Payout Currency*
EUR

Year*
2022

Apply Settings

Seller Overview

+ Add Portlet (16) Export to Excel

Compensation By Month

Compensation For 2022 By Month
From 01/01/2022 To 31/12/2022

Total Compensation

Total Compensation for 2022
From 01/01/2022 To 31/12/2022

Compensation Reference Table

Compensation Plan	Type	Payment Period	Customer(s)	Product(s)	Baseline	Baseline Value	Paid Compensation (...)	Quota
CO-932 New Comp...								
	Growth Absolute Am...	Quarterly	(CustomerId - CD-0...	Quantity		35,749	12.50	[100-23, 1000-50]
		2022-01				22,846	12.50	
		2022-02				3,079	0.00	
		2022-03				9,824	0.00	
		2022-04				0	0.00	

In this section:

- Product Info (Sales Compensation)
- Overview (Sales Compensation)
- Business User Reference (Sales Compensation)
- Admin User Reference (Sales Compensation)
- Technical User Reference (Sales Compensation)
- Release Notes (Sales Compensation)
- Archive of Documentation (SC)

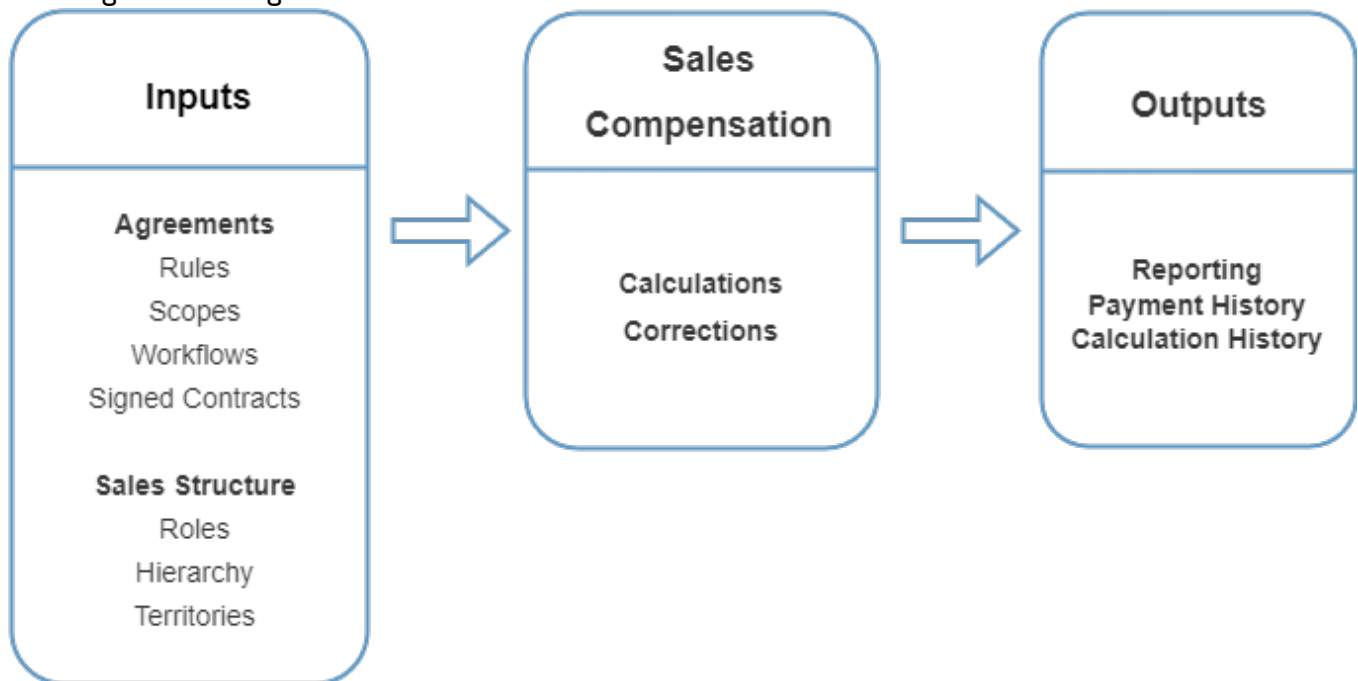
Product Info (Sales Compensation)

The Sales Compensation Package provides a solution for the fast implementation of the company's sales compensation plan. Its purpose is to maintain, control and evaluate execution with the Pricefx application.

Sales Compensation Plan

It outlines the sales agent's base salary as well as the company's commission and incentive program with strategy. Commission, bonus, and incentive structure incentivize sales force to reach their objectives in order to earn a deserved reward. Final pay mix together makes up On-Target Earnings. In other words, the amount of money a salesperson is paid in the end.

Sales compensations encourage positive behaviors in your sales teams that are necessary to achieve your overall organizational goals and results.



Business Roles

The following business roles and individuals participate in the processes surrounding the Accelerator's usage activities.

- **Sales Compensation Administrator**

A person who is responsible for the smooth working of all elements. This person takes the plan from the company's representatives and ensures that individual Sales Managers can create agreements. The person makes sure all salespeople are in the system and individual compensation types and calculation logics that the company uses are present in the system. Sales team changes, adjustments or running plans, contract revisions for a new cycle, or global reporting are activities in their agenda. Typically, this person is from Finance or SalesOps department closely cooperating with Sales Managers, HR, and legal team.

- **Sales Manager**

A person who is responsible for the actual creation of individual Compensation Agreements. The person enters conditions into agreements, negotiates details with Sales Agents, and is able to watch

progress towards set goals. The Sales Manager also generates reports for those who are not able to access the system on their own. It could be an alternative name for Team Lead.

- **Sales Agent**

A person who actually sells and is also the main recipient of compensations. It could be an alternative name for a Sales Representative, SalesRep, or Salesperson. In the context of this accelerator the terms Seller or Payee are also used.

Scope of Usage

The Accelerator is a tool used in agendas of the above-mentioned roles. It helps them simplify, automatize, track or define compensation-related work tasks.

- **Creating Compensation Agreement**

Compensation Agreement is one unit of the main Sales Compensation Plan. It can encapsulate one or more rules, conditions, exclusions, and time spans that are used for the calculation of the final sum for the Sales Agent. This definition can represent a legal contract.

- **Reporting Progress**

Reporting is an activity that can be done on an agreement level or with a scope of several Sales Agents. Provided reports and dashboards are used for the analysis of the progress of the plan.

- **Plan Maintenance**

Corrections of wrongly accounted transactions, new agreement revisions, definitions for a new period of a year, or replacements in the sales team are common activities associated with a well-defined plan.

- **Providing Transparency**

While creating a quote in Pricefx, sellers can see the related compensation amount.

← Demo Deal 2021 🔗 📄 Draft Recalculate Start CW Export Email Quote Duplicate ...

Header **Items** Attachments Workflow Messages Insights

Waterfall for 3 item(s)

Category	Value
List Price	601.3
Sales Channel Discount	-30.1
Promotion Discount	-60.1
Warehousing Adj	-13.0
Packaging Adj	-10.5
Invoice Price	536.7
Net Price	536.7
Warehousing Cost	-12.0
Packaging Cost	-8.1
Pocket Price	516.5
Variable Cost	-233.9
Pocket Margin	282.6
Fixed Cost	-50.8
Gross Margin	231.7

Items + Add Items + Add Folder Mass Edit ... 🔍

Label	Warehousing A...	Packaging Adju...	Net Price	Margin %	Revenue	Price Quality	Pricing
<input checked="" type="checkbox"/> IDPN - circuit breaker - IDPN N - 1P + N	1.61€	1.13€	57.48€	71.99%	57.48€	Show	Indepe
<input type="checkbox"/> Back UPS PRO BN 1350VA, 10 Outlets, 2	11.04€	7.73€	394.05€	42.41%	394.05€	Show	Indepe
<input type="checkbox"/> APC AV C Type 8 Outlet Power Filter, 12	2.38€	1.67€	85.11€	27.27%	85.11€	Show	Indepe

> IDPN - circuit breaker - IDPN N - 1P + N - 13A - ...

Input Parameters **Calculations**

Label	Calculation Result	Message
Lowest Competi...	120.60	€
Sales Compensation		
Sales Represent...	admin	
Quote ID	P-879	
Expected Turnov...	57.48	€
Compensation %	0.00%	
Currency	EUR	
Compensation	0.00	€
Taxes		

Dashboards and Reporting

The ability to accurately track and compare progress towards defined goals is an important aspect of successful strategy execution. The package is distributed with a dashboard and set of reports supporting those processes.

- **Compensation Dashboards**

Interactive reporting in dashboards is used for a deeper understanding of compensation agreements. Users are restricted to see data of their subordinates and their own. You can read more on [Dashboards Description \(Sales Compensation\)](#).

- **Compensation Plan Report**

Here is an example of a basic agreement between the Manager and Sales Agent:



- **Overview Report**

A document summarizing definitions of an agreement with results for each agreement line and total compensation value. This is the one-pager for the agreement in numbers.

- **YTD Report**

It is a simple document showing agreement progress in the current calendar year. It goes into fewer details about each agreement line but focuses on results for the year.

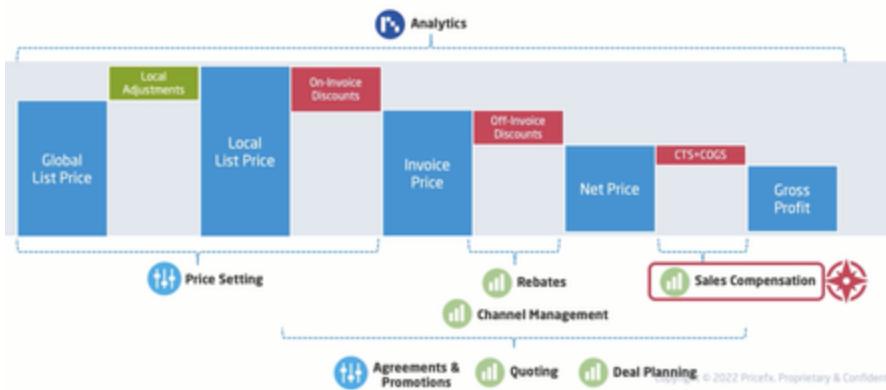
- **Transaction Report**

This report is used for checking all transactions that are included in the calculation of the selected compensation agreement.

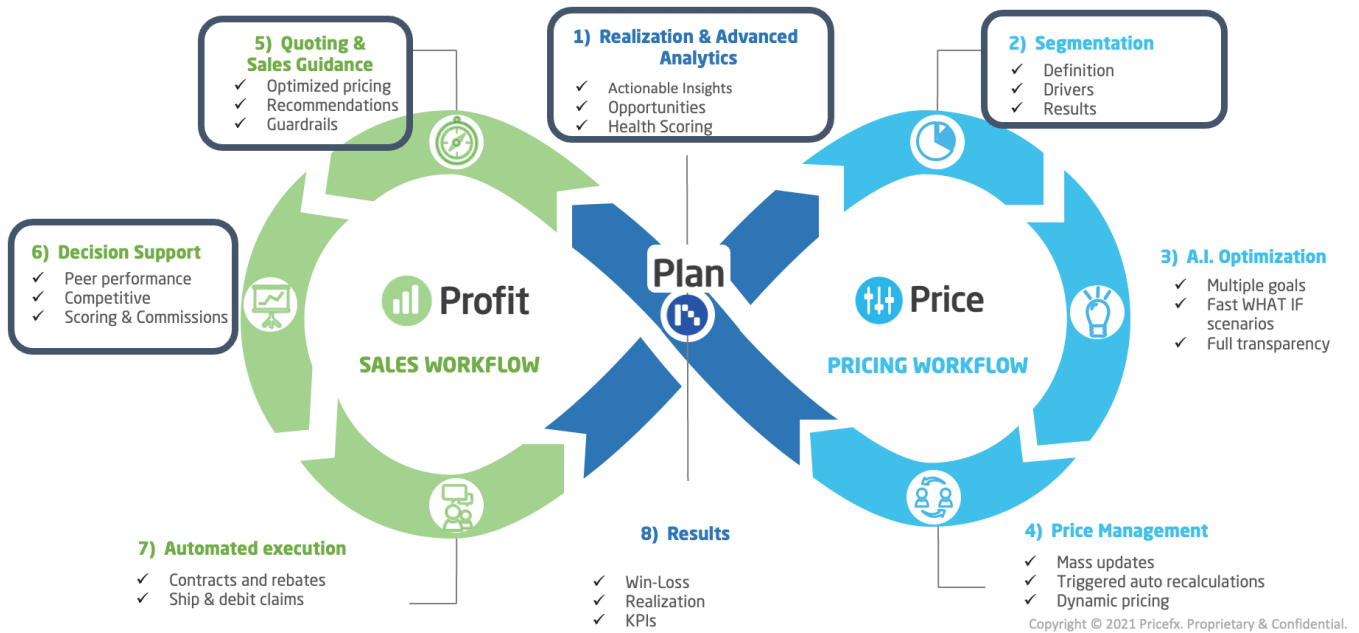
It can be used as an audit supporting tool or document generated when an agreement is fulfilled, see example [SC_Transaction_List_Report_SAMPLE.pdf](#).

Overview (Sales Compensation)

The Sales Compensation 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. This accelerator is designed to provide aggregation at the following point in the waterfall:



Within the design of the Pricefx PLAN/PRICE/PROFIT structure, we can see where the Sales Compensation Accelerator is included:



Business Overview (Sales Compensations)

Premise

You are involved with the management and review of compensation analytics associated with pricing applications as part of a team from either Pricing, Financial, Sales, Sales Ops within your organization.

Desired Outcome

You need to manage your sales compensation agreements and provide the necessary transparency to ensure that sales performance and compensation for individuals are aligned with company goals.

Context and Background

Your organization provides incentives for your internal sales force through sales compensation agreements and these compensations and their impact need to be understood as part of the waterfall process. The amounts and the payouts need to be transparent and well communicated.

Problem

Need for a standard solution to manage sales compensation agreements and a pre-defined set of common compensation types with associated calculation logic. Additionally, analytical dashboards are needed to provide proper insights into year-to-date values and salesperson segmentations.

Solution Capabilities

Once this Accelerator has been implemented and linked to your application, then these components will be available for immediate use:

- Ability to create and revise sales compensation agreements
- Deployment of 8 pre-defined compensation types
- Ability to analyze sales compensation using 4 dashboards
- Creation of approval workflows for sales compensation agreements

Capabilities Summary (Sales Compensation)

Out-of-the-box

Once you install Sales Compensation package and do the initial setup, you will immediately have the option to:

- Create and revise a Sales Compensation Agreement
- Use 8 Compensation Types
- Use 4 dashboards for analysis of progress

Configurable

The following components are flexible and configurable:

- Approval workflow for agreements
- Source of transactional data for calculation of compensation
- Defaults for certain behavior
- Currency symbols

Customizable

If the out-of-the-box capabilities do not suit you, you can further customize:

- Inputs/outputs of provided Compensation Types
- Add new Compensation Types

Business User Reference (Sales Compensation)

- [Sales Compensation Flow \(Sales Compensation\)](#)
- [Dashboards Description \(Sales Compensation\)](#)
- [Glossary \(Sales Compensation\)](#)

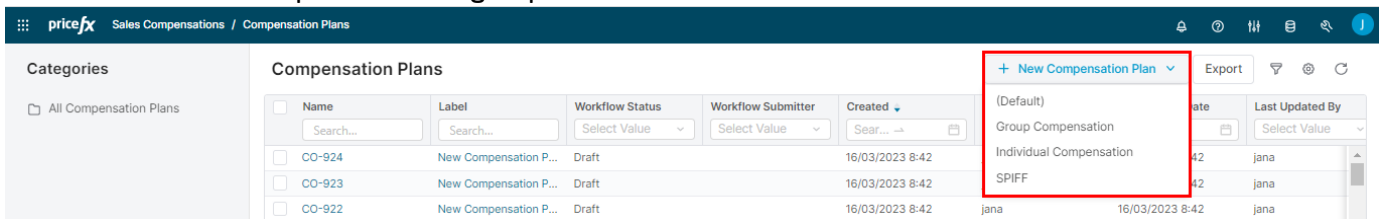
Sales Compensation Flow (Sales Compensation)

This section helps you understand how a sales compensation flow is created and used in the Sales Compensation module.

- [Create Compensation Plan \(Sales Compensation\)](#)
- [Approve Compensation Agreements \(Sales Compensation\)](#)
- [Run Calculation Job \(Sales Compensation\)](#)
- [Verify Compensation Records \(Sales Compensation\)](#)
- [Analyze Results in Dashboard \(Sales Compensation\)](#)
- [Generate Output Documents \(Sales Compensation\)](#)
- [Connect to Third Party \(Sales Compensation\)](#)

Create Compensation Plan (Sales Compensation)

1. Go to **Sales Compensations > Compensation Plan**.
2. Create a New Compensation Plan, using one of these [options](#):
 - a. **Individual Compensation** - Plan for an individual seller.
 - b. **Group Compensation** - Plan for a team where members have the same individual conditions (but there is no team goal to contribute to). Teams also allow you to work with validity (members belong to the team for a defined period of time etc.).
 - c. **SPIFF** (Sales Program Incentive Funds) - Plan for a seller group. This type is targeted for quick and easy creation of typically short-term incentive plans to drive sales of specific goods or for limited time period. Seller groups are based on Master data.



The plan creation consists of the following steps:

1. **Header Level**
 - [Calculated Results](#)
2. **Line Item Level**
 - [User Inputs](#)
 - [Calculated Results](#)
3. **Input Validation**

1. Header Level

On the header level, the package provides standard input entries, summarized line item compensation, holdback value and forecast value, also grouped by Condition Type.

User Inputs (* means it is mandatory):

- **Start Date**^(*) - Enter the start date of the period for the analysis.
- **End Date**^(*) - Enter the end date of the period for the analysis.
- **Customer(s)** - Allows you to choose one or more customers for the analysis.
- Depending on the selected type:
 - **Team**^(*) - Allows you to select a team if it is a group compensation.
 - **Payee**^(*) - Allows you to select a seller if it is an individual compensation.

- **Payee(s)^(*)** - Allows you to select a seller group if it is SPIFF.
- **Payout Currency^(*)** - Allows you to select a payout currency and affects all the calculation values. The currency list displays according to the condition: CcyFrom = DM Currency and CcyValidFrom <= startDate <= CcyValidTo
The values are taken from:
 - Currencies DS - CcyFrom, CcyValidFrom, CcyValidTo
 - Compensation Plan - Start Date
- **Holdback** - Allows you to enter how much of the compensation should be held back from the payee and paid only in the last period of the plan. This mechanism helps you ensure that you do not overpay your Sales Agents in case of cancellations/returns etc. You can configure to take e.g. 20% of compensation from every period and pay it back to the seller in the last period.
The holdback is defined on the header level and is applied separately for every payout (Compensation Record). It is defined in **Holdback Type** as either percentage of a compensation earned in the payment period or by a flat value.
- **Forecast Method** - Allows you to select a Forecast Method to calculate Compensation Forecast:
 - Seasonal Naive
 - Average
- **Sales Goal Increase %** - Input number
- **General Filter** - Allows you to set up a generic transaction data filter.

Calculated Results

Calculated Result show the following details:

- Compensation Detail
 - Previous Compensation
 - Current Compensation
- Forecast
 - Compensation Forecast
 - True-up
 - Compensation Forecast Comparison Chart
 - This Year's Sales Forecast Table
 - This Year's Sales Forecast Chart
- <Grouped by Condition Type>
 - Current Compensation
 - Compensation Forecast

▼ Compensation Detail	
Label	Calculation Result
Previous Compensation	870,791.53
Current Compensation	949,627.82

▼ Forecast	
Label	Calculation Result
Compensation Forecast	448,457.64
True-up	-26,356.27
Compensation Forecast Comparison Chart	Show
This Year's Sales Forecast Table	Show
This Year's Sales Forecast Chart	Show

▼ Flat Bonus	
Label	Calculation Result
Current Compensation	949,627.82
Compensation Forecast	896,915.27

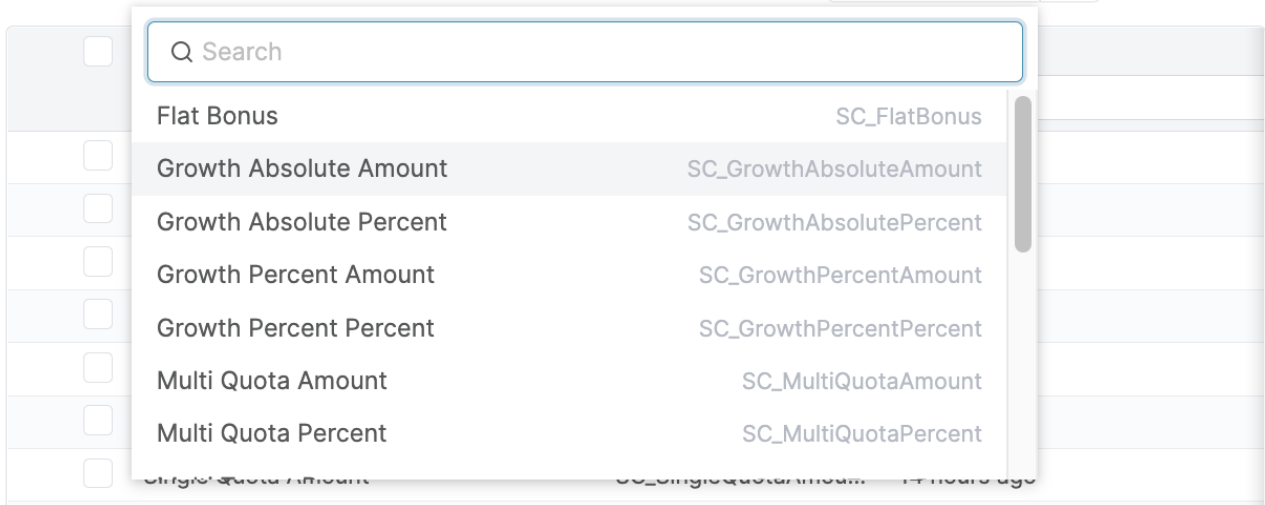
2. Line Item Level

On the line item level, the package provides standard input entries, compensation and forecast value, charts, compensation type info.

1. Go to **Items** tab.
2. Click the **Add Items** button.

Items

+ Add Items



Condition Type	Code
Flat Bonus	SC_FlatBonus
Growth Absolute Amount	SC_GrowthAbsoluteAmount
Growth Absolute Percent	SC_GrowthAbsolutePercent
Growth Percent Amount	SC_GrowthPercentAmount
Growth Percent Percent	SC_GrowthPercentPercent
Multi Quota Amount	SC_MultiQuotaAmount
Multi Quota Percent	SC_MultiQuotaPercent

3. Choose the Condition Type you want to add and enter values at the Line Item Level. There are more than 15 condition types, for details see [Types Overview \(Sales Compensation\)](#).

i When any of the Sellers defined in the plan are missing a Salary (configured in the Seller Salaries SX) in the Payout Currency of the plan, the Variable Pay Linear Amount, Variable Pay Multi Target Amount, and Variable Pay Stepped Amount condition type are hidden from the items selection in the Compensation Plans.


You can set the following inputs:

User Inputs


Flat Bonus

Input Parameters

Customer(s)

Product(s)

Quota *

Target Incentive *

Payment Period *

Quota For

Annual

Deposit Scheme

Non-Cumulative

Baseline Field ?

Invoice Price

- **Customer(s)** - Allows you to choose one or more customers for the analysis. It overrides the customer input on the header level.
- **Product(s)** - Allows you to choose one or more products for the analysis. The inputs are generated based on each condition type you selected on the line item.
- **Compensation Cap** - Allows you to set a limit for the compensation. Only the amount of compensation for a given period which exceeds the cap is paid out. It is applied only on the line item level where each item can have its own specific details - see an example below.
Note: If "Quota For" is set to "Payment Period", this input is set to "Payment Period" and is read only.
- **Payment Period** - Allows you to choose a payment period. The options are:
 - Monthly
 - Quarterly
 - Semi-Annually
 - AnnuallyBy default, it is taken from [SC_AdvancedConfiguration](#).
- **Quota For** - Read-only, taken from [SC_AdvancedConfiguration](#). Quota is minimal amount of sales needed to be eligible for compensation.
- **Deposit Scheme** - Read-only, taken from [SC_AdvancedConfiguration](#). The scheme decides if the calculation is split by the number of periods in the target time span (needed in case Quota For is set to "Plan Duration").
- **Baseline Field** - Read-only, taken from [SC_AdvancedConfiguration](#).




Example of using different caps for line items

Let's assume the following scenario:

- Line Item caps:
 - LI 1: 500 - yearly
 - LI 2: 400 - monthly
 - LI 3: no cap
- Payout months:
 - M1:
 - LI 1 compensation is 400, so no cap applied
 - LI 2 compensation is 100, so no cap applied
 - LI 3 compensation is 500, so no cap applied
 - M2:
 - LI 1 compensation is 150, so a line item cap is applied, because $400 + 150 = 550 > 500$, payed out 100
 - LI 2 compensation is 800, so a line item cap is applied, because $800 > 400$, payed out 400
 - LI 3 compensation is 300, so no cap applied
 - M3:
 - LI 1 compensation is equal to 500, so payed out 0
 - LI 2 compensation is 150, so no cap applied
 - LI 3 compensation is 800, so no cap applied

i In case, you added a line item and then changed some values (Quota For, Deposit Scheme or Baseline Field) in SC_AdvancedConfiguration, the read-only inputs are not updated accordingly. We suggest that you should NOT change this configuration after the deployment but if it is necessary, you need to delete an existing line item and add it again for the update.

Calculated Results

Calculations																			
<ul style="list-style-type: none"> <ul style="list-style-type: none"> <table border="1"> <thead> <tr> <th>Label</th> <th>Calculation Result</th> </tr> </thead> <tbody> <tr> <td> Baseline Value Field Label</td> <td>Invoice Price</td> </tr> <tr> <td>Previous Baseline Value</td> <td>3,627,661.13</td> </tr> <tr> <td>Current Baseline Value</td> <td>4,091,207.21</td> </tr> <tr> <td>Previous Compensation</td> <td>435,395.76</td> </tr> <tr> <td>Current Compensation</td> <td>474,813.91</td> </tr> </tbody> </table> 		Label	Calculation Result	 Baseline Value Field Label	Invoice Price	Previous Baseline Value	3,627,661.13	Current Baseline Value	4,091,207.21	Previous Compensation	435,395.76	Current Compensation	474,813.91						
Label	Calculation Result																		
 Baseline Value Field Label	Invoice Price																		
Previous Baseline Value	3,627,661.13																		
Current Baseline Value	4,091,207.21																		
Previous Compensation	435,395.76																		
Current Compensation	474,813.91																		
<ul style="list-style-type: none"> <table border="1"> <thead> <tr> <th>Label</th> <th>Calculation Result</th> </tr> </thead> <tbody> <tr> <td>Forecast Method</td> <td>Seasonal Naive</td> </tr> <tr> <td>Sales Goal Increase %</td> <td>3.00%</td> </tr> <tr> <td>Forecasted Baseline Value</td> <td>3,736,490.97</td> </tr> <tr> <td>Compensation Forecast</td> <td>448,457.64</td> </tr> <tr> <td>True-up</td> <td>-26,356.27</td> </tr> <tr> <td>Compensation Forecast Comparis...</td> <td>Show</td> </tr> <tr> <td>This Year's Sales Forecast Table</td> <td>Show</td> </tr> <tr> <td>This Year's Sales Forecast Chart</td> <td>Show</td> </tr> </tbody> </table> 		Label	Calculation Result	Forecast Method	Seasonal Naive	Sales Goal Increase %	3.00%	Forecasted Baseline Value	3,736,490.97	Compensation Forecast	448,457.64	True-up	-26,356.27	Compensation Forecast Comparis...	Show	This Year's Sales Forecast Table	Show	This Year's Sales Forecast Chart	Show
Label	Calculation Result																		
Forecast Method	Seasonal Naive																		
Sales Goal Increase %	3.00%																		
Forecasted Baseline Value	3,736,490.97																		
Compensation Forecast	448,457.64																		
True-up	-26,356.27																		
Compensation Forecast Comparis...	Show																		
This Year's Sales Forecast Table	Show																		
This Year's Sales Forecast Chart	Show																		
<ul style="list-style-type: none"> <table border="1"> <thead> <tr> <th>Label</th> <th>Calculation Result</th> </tr> </thead> <tbody> <tr> <td>Quota For</td> <td>Annual</td> </tr> <tr> <td>Deposit Scheme</td> <td>Non-Cumulative</td> </tr> <tr> <td>Payment Period</td> <td>Quarterly</td> </tr> <tr> <td>Payee Name(s)</td> <td>Show</td> </tr> <tr> <td>Customer(s)</td> <td>('Region' = "America")</td> </tr> <tr> <td>Product(s)</td> <td>('ProductClass' = "A")</td> </tr> <tr> <td>General Filter</td> <td>'Region' = "Europe"</td> </tr> <tr> <td>Payout Currency</td> <td>EUR</td> </tr> </tbody> </table> 		Label	Calculation Result	Quota For	Annual	Deposit Scheme	Non-Cumulative	Payment Period	Quarterly	Payee Name(s)	Show	Customer(s)	('Region' = "America")	Product(s)	('ProductClass' = "A")	General Filter	'Region' = "Europe"	Payout Currency	EUR
Label	Calculation Result																		
Quota For	Annual																		
Deposit Scheme	Non-Cumulative																		
Payment Period	Quarterly																		
Payee Name(s)	Show																		
Customer(s)	('Region' = "America")																		
Product(s)	('ProductClass' = "A")																		
General Filter	'Region' = "Europe"																		
Payout Currency	EUR																		

- **Compensation Detail**
- Baseline Value Field Label - Shows the field label to use for calculating the Baseline Value.
- Previous Baseline Value - Total sales for a validity period of the agreement - 12 months.
- Current Baseline Value - Sum of the Current Baseline Value based on the Start Date - End Date.
- Previous Compensation - Compensation value based on sales from the previous period (current contract validity - 12 months) and current quota input conditions.
- Current Compensation - Sum of the current compensation from the current period and current quota input conditions.
- **Forecast**
- Forecast Method
- Sales Goal Increase %
- Forecasted Baseline Value - Sum of the Forecasted Baseline Value
- Compensation Forecast - Sum of the Compensation Forecast
- True-up - Equals to Compensation Forecast - Current Compensation
- Compensation Forecast Comparison Chart
- This Year's Sales Forecast Table
- This Year's Sales Forecast Chart
- **Compensation Type Info**
- Quota For
- Deposit Scheme
- Payment Period
- Payee Name(s)
- Customer(s)
- Product(s)
- General Filter
- Payout Currency

Color coding:

- Previous Compensation: **blue**
- Current Compensation > Previous Compensation: **green**, else **red**
- Compensation Forecast > Current Compensation: **green**, else **red**
- If value does not change, it will be **orange**.

3. Input Validation

The system will show a warning message if:

- Input for the required fields on the header/line level is missing.
- Input for the required field that is less than zero.
- Seller termination date (configured in the Sellers master data) is less than Start Date input on the Compensation Plan.

Sample of input validation:

! InputValidation : INPUT VALIDATION:: Flat Bonus: Target Incentive should be less than Quota.

InputValidation : INPUT VALIDATION:: Single Quota Amount: Quota must be greater than or equal to zero

InputValidation : INPUT VALIDATION:: Single Quota Percent: Compensation % must be greater than or equal to zero

InputValidation : INPUT VALIDATION:: Variable Pay Multi Target Amount: Variable Pay (%) must be greater than zero

Compensation plan calculated with errors

After the details on the header and line item level are filled in and the inputs are validated, the final step is to submit the Compensation Plan.

Approve Compensation Agreements (Sales Compensation)

You can approve/decline Compensation Agreements. This follows the hierarchy based on the sales organization (entered in the Seller Master Table). Once approved/declined, an email confirmation is sent.

The screenshot displays two overlapping windows from the 'pricefx' system. On the left is a 'Submit confirmation' dialog box with a close button (X) in the top right corner. It contains the text 'Do you want to submit the current Quote?' and a text input field for 'Your comment (optional)'. At the bottom are 'Yes' and 'No' buttons. On the right is a larger window titled 'Quote P-714 requires your approval'. It lists details for a 'New Quote' created by 'michel'. The 'Customer Information' section includes Customer ID: CD-00004, Customer Name: Price fx) AG, and Country: Germany. The 'Approval Information' section shows the Reason: 'There is a new Quote need to be approved'. The 'Quote Information' section lists Effective Date: 2019-09-10 and Expiry Date: 2019-09-10. The 'Quote Summary' section provides financial data: Total Revenue: 150 EUR, Total Quantity: 300, Total Margin: -101 EUR, and Total Margin %: -67%. At the bottom of this window are three buttons: 'View Quote' (blue), 'Approve' (green), and 'Decline' (red).

Run Calculation Job (Sales Compensation)

1. Go to **Sales Compensation > Calculations**.



Dashboards

> Pricefx Processes

> Analytics

> Price Setting

> Quoting

> Deal Planning

> Agreements & Promotions

> Rebates & Compensations

✓ Sales Compensations

Compensation Types

Condition Types

Compensation Plans

Calculations



Compensation Records

Accrual Records

Adjustments

> Channel Management

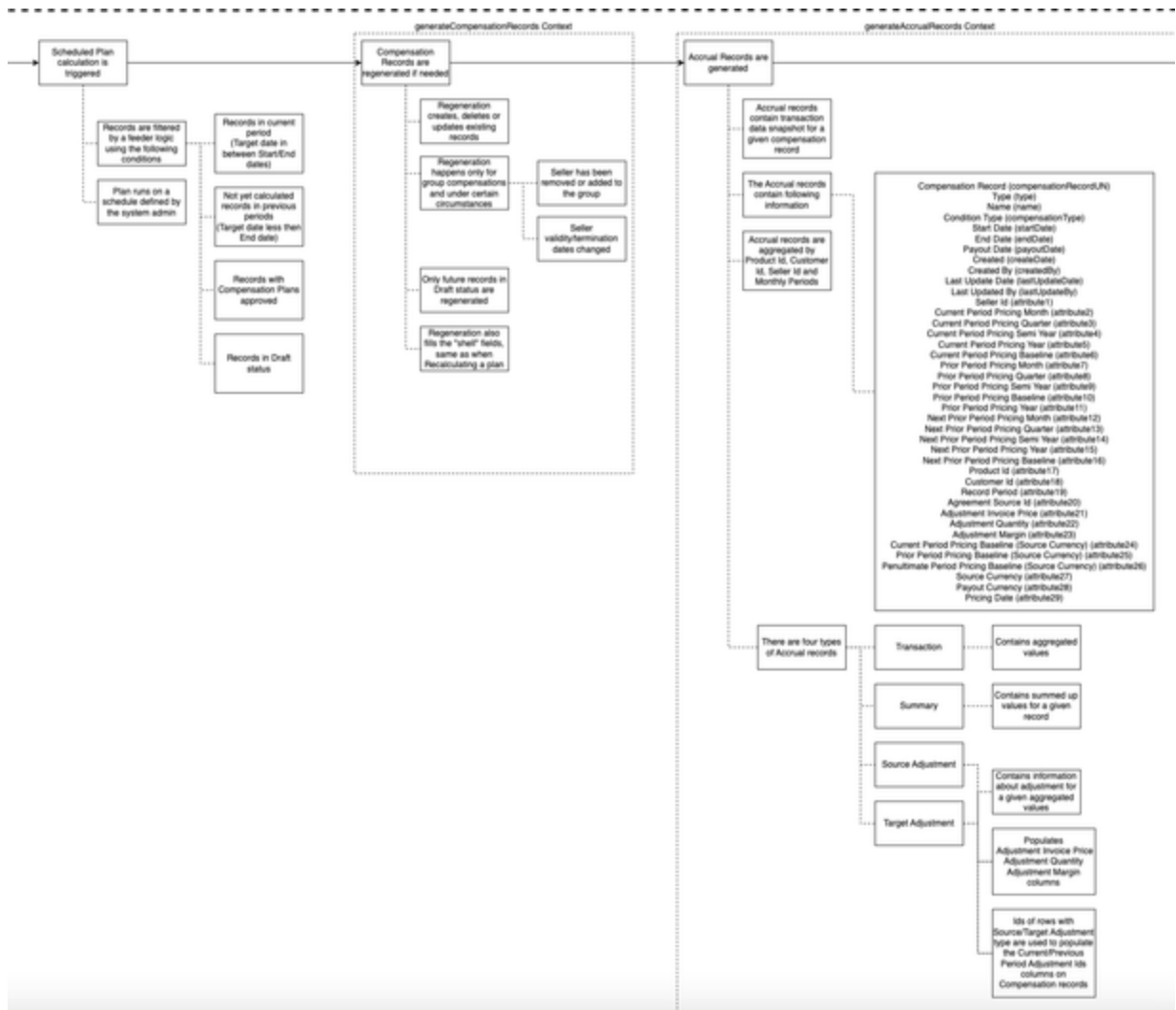
> Optimization

2. Run the below listed jobs . There are scheduled jobs or you can run them manually.

The screenshot shows the 'Calculations' page in the Pricefx system. The page has a header with the Pricefx logo and navigation icons. Below the header is a table with the following columns: Label, Set, Status, Start Date, and End Date. The table contains three rows of data. The first row is 'Sales Compensation Plan', the second is 'Sales Compensation Record', and the third is 'Default'. The first two rows are highlighted with a red box. The 'Sales Compensation Plan' row has a 'Set' of 'Default', a 'Status' of 'Ready', a 'Start Date' of '06/12/2022 7:02', and an 'End Date' of '06/12/2022 7:07'. The 'Sales Compensation Record' row has a 'Set' of 'Default', a 'Status' of 'Ready', a 'Start Date' of '06/12/2022 9:01', and an 'End Date' of '06/12/2022 9:01'. The 'Default' row has a 'Set' of 'Default', a 'Status' of 'Ready', a 'Start Date' of '02/12/2022 18:30', and an 'End Date' of '02/12/2022 18:31'.

Label	Set	Status	Start Date	End Date
Sales Compensation Plan	Default	Ready	06/12/2022 7:02	06/12/2022 7:07
Sales Compensation Record	Default	Ready	06/12/2022 9:01	06/12/2022 9:01
Default	Default	Ready	02/12/2022 18:30	02/12/2022 18:31

- **Sales Compensation Plan** - This job (re)generates Compensation Records and Accrual Records. The results in the Compensation Records are still empty.



- After the job finishes, you can go to the Accrual Records for checking:
 - Go to **Sales Compensation > Accrual Records**.



| Dashboards

- > Pricefx Processes
- > Analytics
- > Price Setting
- > Quoting
- > Deal Planning
- > Agreements & Promotions
- > Rebates & Compensations
- ∨ **Sales Compensations**
 - Compensation Types
 - Condition Types
 - Compensation Plans
 - Calculations
 - Compensation Records

Accrual Records

Adjustments

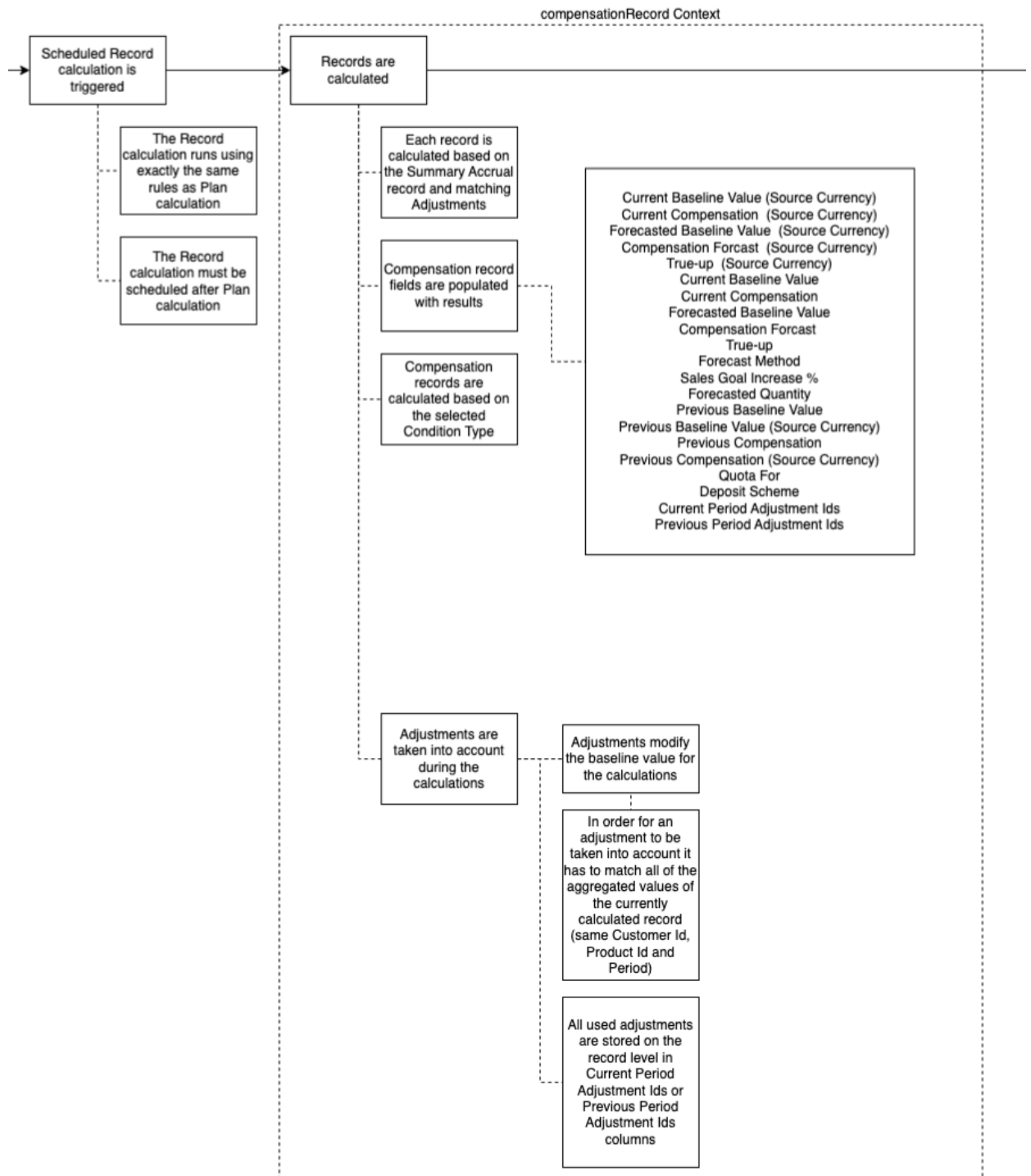
- > Channel Management
- > Optimization

- Filter Agreement Source Id == CO Id && Type == Transaction or Summary you want to check.

The screenshot shows the 'pricefx Sales Compensations / Accrual Records' interface. The table below displays a list of compensation records. The 'Agreement Source Id' column is highlighted with a red box, showing the value 'CO-544'.

Compensation Record	Type	Agreement Source Id	Name	Condition Type	Status	Description	Start Date	End Date	Payout Date	Crez
<input type="checkbox"/>	Search...	Search...	CO-544	Search...	Search...	Search...	Search...	Search...	Search...	Search...
<input type="checkbox"/>	COR-8448	Transaction	CO-544	2022-Q3 Stepped Bo...	SC_SteppedBonus		01/07/2022	30/09/2022	03/10/2022	3
<input type="checkbox"/>	COR-8448	Transaction	CO-544	2022-Q3 Stepped Bo...	SC_SteppedBonus		01/07/2022	30/09/2022	03/10/2022	3
<input type="checkbox"/>	COR-8448	Transaction	CO-544	2022-Q3 Stepped Bo...	SC_SteppedBonus		01/07/2022	30/09/2022	03/10/2022	3
<input type="checkbox"/>	COR-8448	Transaction	CO-544	2022-Q3 Stepped Bo...	SC_SteppedBonus		01/07/2022	30/09/2022	03/10/2022	3
<input type="checkbox"/>	COR-8448	Transaction	CO-544	2022-Q3 Stepped Bo...	SC_SteppedBonus		01/07/2022	30/09/2022	03/10/2022	3
<input type="checkbox"/>	COR-8448	Transaction	CO-544	2022-Q3 Stepped Bo...	SC_SteppedBonus		01/07/2022	30/09/2022	03/10/2022	3
<input type="checkbox"/>	COR-8448	Transaction	CO-544	2022-Q3 Stepped Bo...	SC_SteppedBonus		01/07/2022	30/09/2022	03/10/2022	3

- **Sales Compensation Record** - This job calculates results including the adjustments (if any) in the Compensation Records.



Verify Compensation Records (Sales Compensation)

1. Go to **Sales Compensations > Compensation Records**.



Dashboards

- > Pricefx Processes
- > Analytics
- > Price Setting
- > Quoting
- > Deal Planning
- > Agreements & Promotions
- > Rebates & Compensations

∨ Sales Compensations

Compensation Types

Condition Types

Compensation Plans

Calculations

Compensation Records

Accrual Records

Adjustments

- > Channel Management
- > Optimization

2. Select *Default* in **Set**.

pricefx Sales Compensations / Compensation Records

Compensation Records

Set

Default

ID	Set	Source ID	Source ID w/o revision	Rank	Created
<input type="checkbox"/> COR-6532	Default	CO-393	CO-393	0	13/09/2022 10:50
<input type="checkbox"/> COR-6533	Default	CO-393	CO-393	0	13/09/2022 10:50

3. In the Compensation Records page, you can check the results in Payout Currency and Source Currency of a Compensation Plan that you have created, submitted and had calculated by the Calculations job.

pricefx Sales Compensations / Compensation Records

Compensation Records

Set: Default

Compensation Type	Payee Name	Current Baseline Value	Current Compensation	Pay To ID	Current Baseline Quantity	Current Baseline Value (Source Currency)	Current Compensation (Source Currency)	Previous Baseline Value
Individual Compensation	Seller 003	197,415.06	329.03			1,562.18	2.60	
Individual Compensation	Seller 003	2,013,852.55	3,685.45			15,935.96	29.16	
Individual Compensation	Seller 003	1,901,251.04	6,854.20			15,044.92	54.24	
Individual Compensation	Seller 003	0.00	6,854.20			0.00	54.24	
Individual Compensation	Seller 003	0.00	6,854.20			0.00	54.24	
Individual Compensation	Seller 003	2,696,809.77	11,348.88			21,340.31	89.81	
Individual Compensation	Seller 003	561,749.39	12,285.13			4,445.22	97.21	

For more details about the Calculation Flow see [Flow Diagrams \(SC\)](#).

Analyze Results in Dashboard (Sales Compensation)

See [Dashboards Description \(SC\)](#).

Generate Output Documents (Sales Compensation)

You can export and edit ready-made documents such as:

- Sales Compensation Agreement
- Year-to-date report

SALES COMPENSATION AGREEMENT

PARTIES

- This Compensation Agreement (hereinafter referred to as the "Agreement") is entered into on 2021-01-01 (the "Effective Date"), by and between _____, with an address of _____ (hereinafter referred to as the "Manager") and Tareem Levy with an address of _____ (hereinafter referred to as the "Sales Agent") (collectively referred to as the "Parties").
- Whereas, the Sales Agent agrees not to pursue the selling of a competing product for any competitor during the term of this Agreement.

AGREEMENT

- The Parties hereby agree that the Manager authorizes the Sales Agent to sell _____ on behalf of the Manager and the Sales Agent agrees to indemnify himself/herself as the authorized Sales Agent of the Manager for such sale.

THE SALE

- The Parties agree that the prices of the product will be set by the Manager, that the Manager will obtain and provide the promotional materials (if any) for the Sales Agent to use and to obtain sales, and the Manager will provide a Sales Agreement to the Sales Agent for the Sales Agent to use in the sale of the product.

TERM

- This Agreement shall be effective on the date of signing this Agreement (hereinafter referred to as the "Effective Date") and will end on 2021-12-31.
- The term of this Agreement may be extended upon the provision of written consent from both Parties.

GEOGRAPHY

- The Parties agree that the Sales Agent will sell the product in the following geographical area/territory _____ and will not sell the product anywhere else.

COMMISSION EARNINGS

- The Parties hereby agree that the Sales Agent will be paid by the Manager by the following means:

Compensation 8-241 - 2021-Levy Year-To-Date Report

Start Date: 2021-01-01 End Date: 2021-12-31
Created By: My page Created Date: 2021-11-11
Approved By: My page Sales Agent: Tareem Levy

Qty	Label	Type	Commission	Product	Amount	Balance	Commission
1	SC Manager	SC Manager		Quantity	0		0
1	Plan	Plan					0
Total Compensation:							0

Connect to Third Party (Sales Compensation)

You can pay out sales reps by connecting to a third party (e.g. ERP) solution. To do so, connect with your existing payroll solution and automatically send compensation amount to be paid on a regular basis.

Dashboards Description (Sales Compensation)

Sales Compensations Accelerator has four dashboards used for overview, analysis and tracking of compensations:

- [Admin YTD Summary](#) - Overview dashboard for tracking all payouts for SC administrators.
- [Payouts and Plans](#) - Personal dashboard for sales agents, managers and administrators where they see planned and past payouts together with their plans.
- [Seller Overview](#) - Provides a progress overview for individual sales agents for the current year.
- [Transactions and Statistics](#) - Analytical dashboard for sales agents, managers and administrators that can be used for drilling down to transaction level while respecting the individual permissions.

i All dashboards (as of version 1.6.0) show results relevant for Approved and/or Superseded Compensation Plans and for Compensation Records of any status, unless clearly visible and stated otherwise.

Dashboards Visibility

The dashboards are visible to the following user roles:

- **Admin YTD Summary** - For administrators only. No further limitations.
- **All other dashboards** - For administrators, managers and agents.
 - Admins can see everything.
 - Managers can see data related to them and their subordinates.
 - Agents can see the data related to them.

i The dashboard access is controlled by Sellers records from the Sellers master data. If you see the below message, it means your Pricefx user ID is not associated with any record:

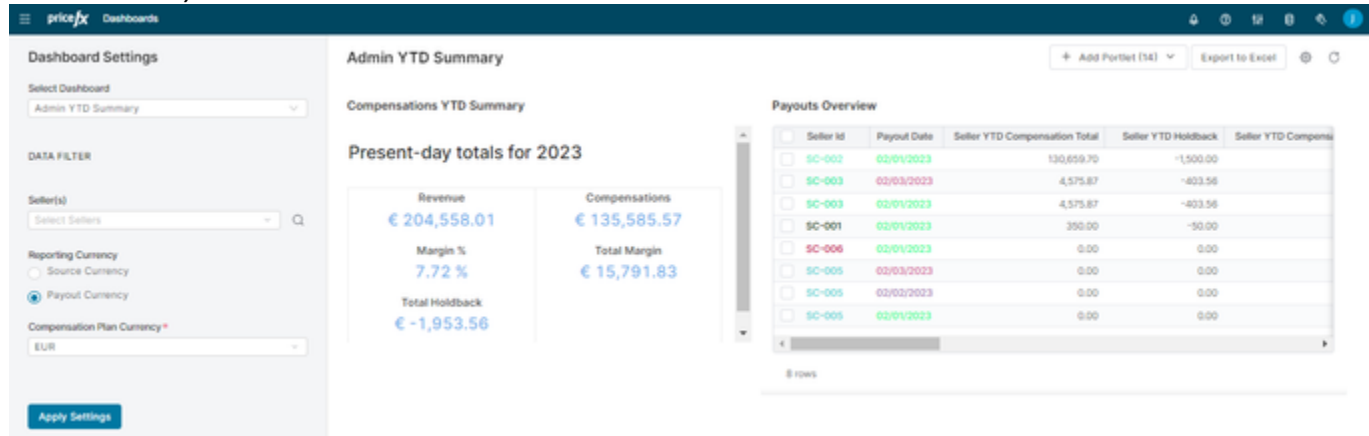
Sales Agent: "There are no Compensation Records associated with your seller data (your user account: %s)"

Sales Manager: "There are no Compensation Records associated with your seller data or that of your subordinates (your user account: %s)"

Admin YTD Summary (Sales Compensation)

This dashboard is designed to provide a quick year-to-date overview for administrators. It shows "where we are" for a current year regarding Sales (Revenue) vs. Compensations paid to different Sellers. You can also drill down if you want to find more details on a specific Payout or Seller.

i Access to this dashboard is restricted to users with the user group SC_Administrator (label: [SC] Administrator).



Dashboard Settings

You can set the following fields:

- **Seller(s)** – Allows you to select one or more Sellers. If left empty, the whole salesforce is selected.
- **Reporting Currency**
 - If **Source Currency** is selected, the dashboard works with just pre-currency changes (no filtering, all results in Datamart currency).
 - If **Payout Currency** is selected, an additional option is displayed:
 - **Compensation Plan Currency** – Allows you to select a currency from those specified as "Payout Currency" in plan(s) of the selected Seller(s). Mandatory field.

Note:

- Only "Approved" or "Superseded" Compensation Plans are taken into account.
- Only CORs (in any state) for Plans in "Approved" or "Superseded" status are taken into account.

Portlets

Compensations YTD Summary (Payout Summary Portlet)

Shows sums from the beginning of the year until now for the selected Seller(s), covering all their Compensation Plans. The sums are for:

- **Revenue** – Sum of all transactional revenue (YTD sales in the pre-changes Payout table, without any aggregation).
- **Compensations** – Sum of YTD compensations for all payouts from the Payout table.

- **Margin %** - Equals to "Total Margin / Revenue" (similarly to YTD Margin % in the pre-changes Payout table, without any aggregation).
- **Total Margin** - Sum off all margins (YTD Margin in the pre-changes Payout table, without any aggregation).

Payouts Overview (Payout Table Portlet)

Provides a high level overview with a split between individual, group and SPIFF compensations. It has information on when the respective line was or will be paid.

Numbers with a yellow background are incomplete summaries for the next month. Until their end date is finalized, their value can change.

- **Seller ID** - Stands for Seller ID of a given COR. CORs are filtered for this year but the payout date has to be earlier than today. The currency filter stays as is. Each Seller has a color assigned (in this column only). It is a different color than in the Payout Date column.
- **Payout Date** - Stands for the Payout Date of a given COR. Each Payout Date has a color assigned (in this column only). It is a different color than in the Seller column.
- **Seller YTD Compensation Total** - Sum from CORs (compensation) for a given Seller. If the Seller has more Payout Dates, the same sum is listed on every row.
- **Seller / Payout Date Compensation Total** - Sum from CORs (compensation) for given keys (Seller and Payout Date).
- **Baseline** columns of various kinds (e.g. Revenue Baseline, Margin Baseline etc.). There are as many "baseline columns" as there are different baseline fields (used for the whole table). Value: Sum of the baseline from CORs of a given "type" (Revenue + Revenue, Margin + Margin etc.).
- **Individual Compensation** - Contribution to "Seller / Payout Date Compensation Total" of Individual plans for the given keys (Seller, Date).
- **Group Compensation** - Contribution to "Seller / Payout Date Compensation Total" of Group plans for the given keys (Seller, Date).
- **SPIFF Compensation** - Contribution to "Seller / Payout Date Compensation Total" of SPIFF plans for the given keys (Seller, Date).

Drill-down on Details

It is possible to drill down to Payout or Seller details from this dashboard. In Payouts Overview, the information is grouped by Seller and Payout Date. When you select one or more rows, there are additional options displayed below the table:

Payouts Overview

<input type="checkbox"/>	Seller Id	Payout Date	Seller YTD Compensation Total	Seller YTD Holdback	Seller YTD Compensation Payment	Seller / Payout Date Cor
<input type="checkbox"/>	SC-002	02/01/2023	130,659.70	-1,500.00	132,159.70	
<input type="checkbox"/>	SC-003	02/03/2023	4,575.87	-403.56	5,707.25	
<input type="checkbox"/>	SC-003	02/01/2023	4,575.87	-403.56	5,707.25	
<input type="checkbox"/>	SC-001	02/01/2023	350.00	-50.00	1,477.60	
<input type="checkbox"/>	SC-006	02/01/2023	0.00	0.00	0.00	
<input checked="" type="checkbox"/>	SC-005	02/03/2023	0.00	0.00	0.00	
<input checked="" type="checkbox"/>	SC-005	02/02/2023	0.00	0.00	0.00	
<input checked="" type="checkbox"/>	SC-005	02/01/2023	0.00	0.00	0.00	

3 selected item(s) Payout Date Details Payout / Seller Details Seller Yearly Overview Clear selection

The context buttons are:

- **Payout Date Details** - Takes you to the Payouts and Plans dashboard with the following inputs:
 - Seller(s) - Unselected
 - Reporting Currency - Source Currency
 - Payout Year - Year of the payout on the selected row
 - Payout Date - Specific date of the payout on the selected row
- **Payout / Seller Details** - Takes you to the Payouts and Plans dashboard with the following inputs:
 - Seller(s) - Specific seller on the selected row
 - Reporting Currency - Payout currency
 - Compensation Plan Currency - Specific currency on the selected row
 - Payout Year - Year of the payout on the selected row
 - Payout Date - Specific date of the payout on the selected row
- **Seller Yearly Overview** - Takes you to the Seller Overview dashboard with the following inputs:
 - Seller(s) - Specific seller on the selected row
 - Compensation Plan - Unselected
 - Payout Currency - Optional, depending on whether it was specified
 - Year - Year of the payout on the selected row

Payouts and Plans (Sales Compensation)

This dashboard is a dedicated overview of Sales Agents' Compensation Plans and Payouts. The dashboard provides a list of (Approved or Superseded) Compensation Plans and their structure without having to access the Sales Compensations module. Additionally, users can also see their payouts (made or planned).

The dashboard is intended to be used by Sales Agents, Sales Managers as well as Sales Compensations administrators. Sales Agents can see only their own data, Sales Managers can see details for all their reports (based on the "Reports To" field in the Seller Master table) and SC Admins have unlimited selection.

The screenshot shows the 'Payouts and Plans' dashboard. On the left is a 'Dashboard Settings' sidebar with filters for Seller(s), Reporting Currency (Source or Payout), Payout Currency, Payout Year, and Payout Date. The main area is divided into three sections: 'Payout Summary' (Summary 03/04/2023) showing key metrics, 'Active Compensation Plans' (a table with columns: Seller Id, Seller Name, Label, Name, Type), and 'Payouts' (a table with columns: Seller Id, Seller Name, Label, Source Id, Header Type, Payout Date, Compensation Total, Holdback, Compen).

Dashboard Settings

You can set the following fields:

- **Seller(s)** - Allows you to select a Seller group. If left empty, the whole salesforce is selected (based on permissions). It is pre-selected and read-only for Sales Agents. Only reporting agents can be selected by Sales Managers.
- **Reporting Currency**

- If **Source Currency** is selected, the dashboard works with just pre-currency changes (no filtering, all results in Datamart currency).
- If **Payout Currency** is selected, an additional option is displayed:
 - **Compensation Plan Currency** - Allows you to select a currency from those specified as "Payout Currency" in plan(s) of the selected Seller(s).
- **Payout Year** - Shows only years where there were payouts for the selected Seller(s). By default, the current year is selected.
- **Payout Date** - Shows only dates of a selected Payout Year where there were payouts for the selected Seller(s). By default, the payout date closest to the current date (in future) is selected.

Portlets

Payout Summary

Quick summary of all payouts that have been paid in the selected period. The following details are displayed:

- **Revenue** as a sum of revenues from transaction data based on the selected payout date and currency input.
- **Selected Sellers**
- **Unique Active Compensation Plans**
- **Payouts**
- **Sellers with Active Plans**
- **Active Plan / Seller Combinations**

Active Compensation Plans

Table showing a list of all open plans. The following details are displayed:

- **Seller ID**
- **Seller Name**
- **Label** of the Compensation Plan
- **Name** of the Compensation Plan
- **Type** of the Compensation

Clicking a Compensation Plan opens a detail. In the detail view, the user can see a simplified version of the configuration from the Compensation Plan stored in the Sales Compensations module. The view does not provide current progress but shows the definition on one screen when a line is clicked.

Payouts

This table gives an overview of what has been paid and what is estimated for the next scheduled payment.

Each Compensation Plan can have several payouts because they are linked to the conditions (items) of the plan. The user is able to see how much and when payouts occurred in past.

Numbers with a yellow background are incomplete for the upcoming months. Until their due date is reached, their value can change.

The following details are displayed:

- **Seller ID**
- **Seller Name**
- **Label** of the Condition Types
- **Source ID** of the Compensation Plan

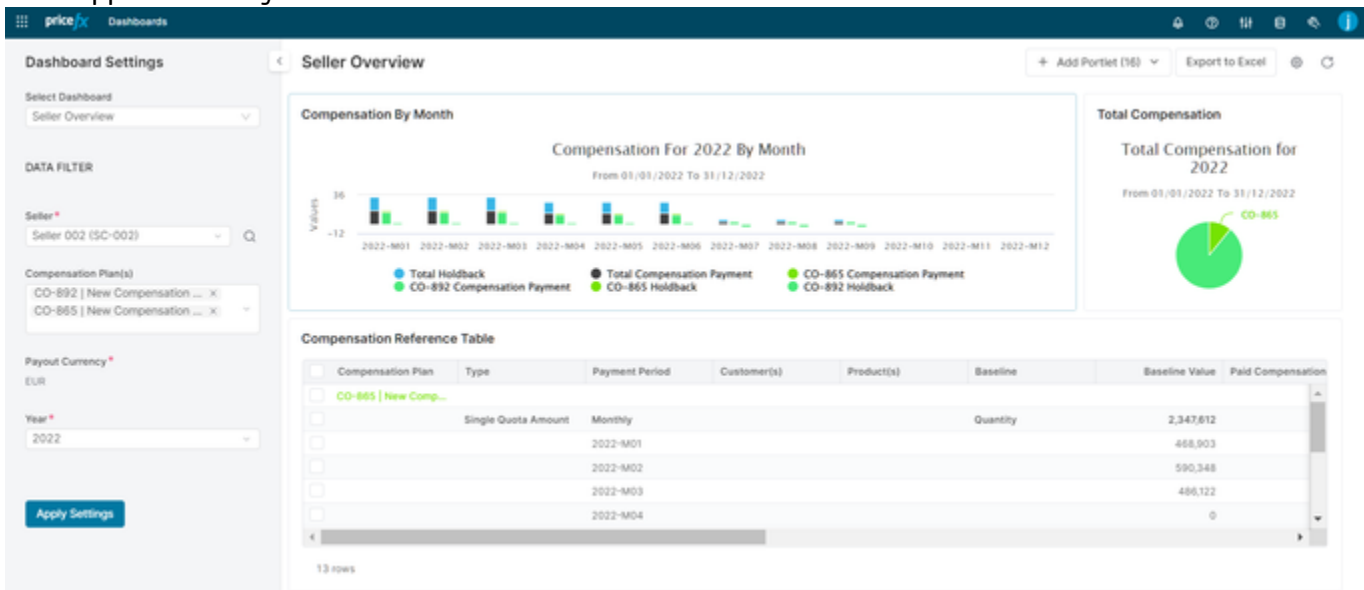
- **Header Type** representing the Compensation Type
- **Payout Date**
- **Compensation Total**
- **Holdback** representing the amount that will be paid in the last period
- **Compensation Payment** calculated as: Compensation Total - Holdback
- **Currency**

When you select one or more rows in the table, there is a context button **Details** at the bottom of the table. This button opens the Transactions and Statistics dashboard with some of the context already prefilled.

Seller Overview (Sales Compensation)

This dashboard provides a progress overview for a particular Sales Agent for the current year. A central focus is to show income in a form of a table and bar charts for Compensation Plans or time periods (months, quarters, ...).

Its primary use is for Sales Agents but it will give valuable insights to Sales Managers or SC Administrators as a supportive analytical tool.



Dashboard Settings

You can set the following fields:

- **Seller** - Allows you to select a Sales Agent. The Compensation Plan input is affected by the selected value. For a given Sales Agent, this field is pre-filled. Sales Managers can see their reports and themselves.
- **Compensation Plan(s)** - Allows you to select one or more customer specific Compensation Plans. If left empty, all agreements assigned to the agent are compared. Only Approved or Superseded Compensation Plans are available here.
- **Payout Currency** - This field is pre-filled and read only if there is just one value - all plans are paid out in the same currency. If more currencies are used, this field allows you to select one.
- **Year** - This field is pre-filled and read only if there is just one value - the selected plan was valid for one year for a given seller. For plans that span multiple years (start/end date) you can select more values. (This can be handy if you want see the history, compare previous years with the current one etc.) When a current year is selected, the dashboard shows Year To Date values.

Portlets

Year To Date Compensation By Time Period

This portlet is labelled as **“Compensation by Month”**.

It is a stacked column chart aggregating all compensations for the selected Sales Agent in time periods. If no specific Compensation Plan was selected, one bar combines all Compensation Plans together giving a total compensation sum that was paid out to the Sales Agent.

Year To Date Compensation By Compensation Plan

This portlet is labelled as **“Total Compensation”**.

It is a pie chart showing the contribution of compensations in individual plans.

Year To Date Compensation Plan Table

This portlet is labelled as **“Compensation Reference Table”**.

It is a summary table showing the breakup of Compensation Plans through Line Items (conditions) down to Compensation Record in a single table. A user can see which lines or periods are outliers.

When you select one or more rows in the table, there are two context buttons at the bottom of the table:

- **Transaction Details** which opens the Transactions and Statistics dashboard with some of the context information already prefilled.
- **Plan Details** which opens details of the Compensation Plan which was selected in the table or which belongs to the selected Compensation Records.

Transactions and Statistics (Sales Compensation)

This dashboard provides an overview of all transactions included in the calculation of the selected Compensation Plan with the possibility to further filter on specific compensation record or transaction date.

The dashboard is for Sales Agents, Sales Managers as well as Sales Compensations administrators. Sales Agents can see only their own data, Sales Managers can see details for all their reports (based on the “Reports To” field in the Seller Master table) and SC Admins have unlimited selection.

Transaction Statistics

No. of Transactions	No. of Sales Agents
41	1
No. of Unique SKUs (Products IDs)	No. of Unique Customers
15	1
Total Quantity	No. of Negative Quantity Transactions
1,816	0

Transaction List

Date	Seller ID	Seller Name	Customer ID	Product ID	Quantity
01/09/2022	SC-002	Seller 002	CD-00048	MB-0029	10.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0027	65.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0036	62.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0017	68.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0038	4.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0009	47.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0035	30.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0028	48.00
01/09/2022	SC-002	Seller 002	CD-00048	SV-0005	34.00
01/09/2022	SC-002	Seller 002	CD-00048	SV-0006	41.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0053	66.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0036	30.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0001	40.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0038	22.00
01/09/2022	SC-002	Seller 002	CD-00048	MB-0035	38.00

Compensation Records

ID	Name	Label	Seller ID	Seller Name	Compensation Plan ID	Compensation Plan	Compe
CDR-11997	2022-Q3 Single Quota Amount-SC-002	Single Quota Amount	SC-002	Seller 002	CD-892	New Compensation Plan	Individ

Dashboard Settings

You can set the following fields:

- **Seller(s)** - Allows you to select a Seller group. If left empty, the whole salesforce is selected (based on permissions). It is pre-selected and read-only for Sales Agents. Only reporting agents can be selected by Sales Managers.
- **Compensation Plan Currency** - Allows you to select a currency from those in which plans were created. If a currency is selected, it limits the possible selection in the other inputs. If nothing is selected, the dashboard uses Datamart Source Currency.
- **Compensation Plan Type(s)** - Allows you to select one or more Compensation Plan Types: Individual, Group or SPIFF. If left empty, all types are selected.
- **Compensation Plan Status(es)** - Allows you to select one or more Compensation Plan States. If left empty, all states are selected. See also a note below.
- **Compensation Plan(s)** - Allows you to select one or more Compensation Plans for a sales person of the selected Compensation Plan Type. If left empty, all plans are selected.
- **Compensation Record(s)** - Allows you to select one or more Compensation Records associated with a given Compensation Plan. If left empty, all records are selected.
- **Date from/to** - Defines time range for the dashboard. If left empty, there is no date filtering.

Note on Compensation Plan Status

- **Sales Agents:** They can see ONLY statuses "Approved" and "Superseded" if those are applicable for "their" plans. They cannot delete all possible values (Approved and Superseded if any).
- **Sales Managers** (as they have *Manage Plans* user role) **and Administrators:** They are NOT limited to any selection and they can leave the field blank which acts as "no filter".
- The input is pre-populated to "Approved" by default for all roles.

Portlets

Transactions Statistics

Statistical overview of the selected scope from Compensation Plans in a form of several tiles. The following information is represented:

<ul style="list-style-type: none">• Number of Transactions• Number of unique SKUs (Products IDs)• Total <Baseline>	<ul style="list-style-type: none">• Number of Sales Agents• Number of unique Customers• Number of Negative <Baseline> Transactions
--	--

Note: There are as many Baseline rows in the tile as there are different baseline fields in the scope, e.g. Quantity, Invoice Price.

Transaction List

A list of selected columns from source transactions that were used as calculation baseline for the selected scope.


- Date
- Seller ID
- Seller Name
- Customer ID
- Product ID
- Baseline - There are as many Baseline columns as there different Baseline Types in all the CORs listed in this table.

Compensation Records

Provides information about attributes for Compensation Records. It can contain just one line if the analysis is focused on one specific condition from the Compensation Plan.

The "ID" column is an active link leading directly to the full detail of the Compensation Record, similarly for "Compensation Plan ID" which leads to the full detail of the Compensation Plan. The SC Administrator can follow the link and see a context that was not available on the dashboard. An example is "Calculation Baseline" showing raw transaction data for the selected Compensation Record.

- ID of the Compensation Record
- Name of the Compensation Record
- Label of the Compensation Record
- Seller ID
- Seller Name
- Compensation Plan ID
- Compensation Plan
- Compensation Plan Type

 This table is affected by the Date From/To Filters, which means that if there is an overlap of the record validity dates with the currently defined dates, then the record will be displayed in the table.


 Tip: Use Group By option on a column you would like to aggregate.

Glossary (Sales Compensation)

This page lists specific terms, vocabulary, and definitions that are used in the context of the Sales Compensation package.

Term	Description
Baseline	Refers to the Baseline field in Transactions Datamart used as a baseline for calculation, e.g. Net Price or Quantity.
Baseline Value	Value of the specified baseline (usually sum etc.), e.g. 5000 (EUR), or 1000 (pieces).
Commission	Always in cash form, commission is an income payment. It could be a percentage of a product or service sold.
Compensation	A term used in the system as a final reward. In reality, it could be for some companies just a bonus, for other commission and in some cases incentives.
Incentive	An amount of money or non-monetary reward to motivate someone to achieve something.
On-Target Earnings (OTE)	Refers to an employee's pay mix made of basic salary and the additional variable component such as commission as their compensation.
Pay mix	The ratio of base salary to target compensations that make up On-Target Earnings (OTE).

Quota	Minimal amount of sales needed to be eligible for compensation.
Sales Agent	A person who actually sells and is also the main recipient of compensations. It could be an alternative name for a Sales Representative, SalesRep or Salesperson.
Sales Compensation Administrator	A person who is responsible for the smooth working of all elements. This person takes the plan from the company's representatives and ensures that individual Sales Managers are able to create agreements. The person makes sure all people are in the system and individual compensation types and calculation logics which the company uses are present in the system. Sales team changes, adjustments or running plans, contract revisions for a new cycle or global reporting are activities in their agenda. Typically, this person is from Finance or SalesOps department closely cooperating with Sales Managers, HR and legal team.
Sales Compensation Plan	It outlines sales agent' base salary as well as the company's commission and incentive program with strategy. Commission, bonus and incentive structure incentivize sales force to reach their objectives in order to earn a deserved reward. In other words, sales compensation itself is the amount of money a salesperson is paid in the end. Sales compensation encourages the positive behaviors in the sales teams that are necessary to achieve the overall organizational goals and results.
Sales Manager	A person who is responsible for the actual creation of individual Compensation Agreements. The person enters conditions into agreements, negotiates details with Sales Agents and watches progress towards set goals. Sales Manager can also generate reports for those who are not able to access the system on their own. It could be an alternative name for Team Lead.
SPIFF	Sales Program Incentive Funds. In this accelerator, it refers to one of the Compensation Types and it allows you to select (as a payee) a seller group from Master data.
True Up	Also called Catch Up. Refers to an adjustment made to a previous estimate to match it with the actual figures.

 See also the general [glossary](#).

Admin User Reference (Sales Compensation)

- [Mandatory Data \(Sales Compensation\)](#)
- [Installation \(Sales Compensation\)](#)
- [Configuration \(Sales Compensation\)](#)
- [Advanced Configuration \(Sales Compensation\)](#)
- [Upgrade \(Sales Compensation\)](#)
- [Troubleshooting \(Sales Compensation\)](#)
- [Known Limitations \(Sales Compensation\)](#)

Mandatory Data (Sales Compensation)

Type	Data	Fields	Use Case
------	------	--------	----------

Master Data	Product	<ul style="list-style-type: none"> Product ID Optional: Up to 30 custom attributes 	There are various Product Group inputs throughout the solution that require this data, e.g. during plan creation or adjustments.
	Customer	<ul style="list-style-type: none"> Customer ID Optional: Up to 30 custom attributes 	There are various Customer Group inputs throughout the solution that require this data, e.g. during plan creation or adjustments.
	Seller	<ul style="list-style-type: none"> Seller ID Seller Name Reports To Pricefx User Account ID <p>Optional:</p> <ul style="list-style-type: none"> First Name Surname Active Termination Date Hire Date Region Market Status 	Seller data is essential for Sales Compensations core functionality.
Transaction Data	Sales Data (Transactions)	<ul style="list-style-type: none"> Baseline Value (in most cases, it will be Revenue /Invoice Price) (Money) Invoice Price (Money) Quantity (Quantity) Margin (Money) Seller Id (String) Product Id (String) Product Name (String) Customer Id (String) Pricing Date (Date) Currency (String) 	Transactions data is essential for Sales Compensations core functionality.

Installation (Sales Compensation)

This tutorial will guide you through the installation of the Sales Compensation Accelerator.

In this section:

- [Prerequisites](#)
- [Installation Steps](#)
 - [Select Partition for Deployment](#)
- [Post-installation Steps](#)
 - [Add Indexing](#)
 - [Enable React UI](#)
 - [Schedule Plan Calculation Task](#)
 - [Schedule Record Calculation Task](#)
 - [Assign Access to Users](#)

Prerequisites


Before you start, ensure that you have:

- Access to a partition on the Pricefx server (9.0 or newer). You will need:
 - Server URL
 - Partition name
 - Username and password for a partition user with sufficient rights for using the Accelerator
 - License on the partition must cover the Analytics and Sales Compensations modules
- Access to Pricefx PlatformManager
 - Username and password for PlatformManager user
 - The user must have the following permissions for your partition (to which you plan to deploy the Accelerator):
 - Permission *Marketplace Templates - deploy*
- [Mandatory data](#), including Transaction data with a specific Datamart structure

The package contains several components and their technical description is provided on the [Components \(Sales Compensation\)](#) page.

Installation Steps

Select Partition for Deployment

1. In PlatformManager, navigate to **Marketplace > Accelerator Packages**, find the *Sales Compensation*.
2. Click **Deploy** and select a partition to which you want to deploy the package.
3. Click **Deploy**.
4. A warning dialogue will appear. After you read the warning text and you agree with the conditions, you can click **Continue**.
 -  If you need to leave the deployment process before it is finished, you can always come back later. The wizard will offer you to either start again, or continue in the previously started process.
5. Set up Datamart mapping of required fields from Datamart and default values for a few parameters.

Settings

Source Type *

Source Name

Baseline Value *

Invoice Price *

Quantity *

Seller Id *

Customer Id *

Product Id *

Product Name *

Pricing Date *

Margin *

Currency *

Currency ▼

Payout Days

4

Payment Period

Quarterly ▼

Quota For

Annual ▼

Deposit Scheme

Cumulative ▼

Continue Cancel

- Upload the Seller Master data and define the mapping. See also a note on Seller Master data behavior during deployment in [Configuration \(Sales Compensation\)](#).

Data Mapping

Sample from your uploaded DataExport-1656008082678.csv file. Your file contains 7 lines:

Seller Id String	Seller Name String	Reports To String	First Name String	Surname String	Pricefx User Account Id String	Active Boolean ▼
SC-001	Seller 001		Grover	Oranger	jakub	false
SC-002	Seller 002	SC-001	Levy	Anje	giang1	false
SC-003	Seller 003	SC-002	Anieta	Bauer	michal	false

Parsing Options

Separator ^{*} : (Semicolon) ▼ Quote character : Escape character : \ Decimal Separator : . Date Format : yyyy-MM-dd

Continue Cancel

- The deployment is complete. Go to your partition and continue with the manual steps required after deployment.

Post-installation Steps

After the package is deployed to your partition and all automatic installation steps are done, you need to do a few manual tasks before you start with configuration and package adjustments to your specific business needs.

Add Indexing

To avoid performance issues when running Calculation Tasks, indexing for COARs on the partition should be added, for these fields: accrualType, attribute1, attribute20.

Example command:

```
MariaDB [pricafx_templates_dev]> create index typeAttr1Attr20 on
CompensationAccrualRecord(partition_id, accrualType, attribute1, attribute20)
using BTREE;
```

Enable React UI

The package is designed to work with the latest UI engine in the Sales Compensations module. For proper functionality is necessary to enable it manually.

1. From the **Administration** menu go to the **Feature Flags** section.
2. Activate the following Feature Flags:
`dashboard.useReactDashboard`
`useReactFor.advancedFilter`
3. Log out and log back in for the changes to take effect.

Schedule Plan Calculation Task

In this step, you will create a task for the plan calculation scheduler.

1. Go to **Sales Compensations > Calculations**.
2. Add a new calculation.
3. Enter a label, e.g. 'SC_Calculation'.
4. Select a set 'Sales Compensations' and click the **Add** button.
5. Click the label of the newly added line.
6. Go to the Calculation tab.
7. As Logic/Formula choose '[SC] Sales Compensation'
8. As Feeder Formula choose '[SC] Compensation Record Calculation Feeder'.
9. Set the Calculation Type to Plan Calculation Task.
10. Optionally, enter StartDate and EndDate.
i Leave empty unless you need to calculate some specific time frame.
11. Click **Save** in the left upper corner.
12. Enable **Incremental** calculation on the left side of your screen.
13. Schedule the calculating task according to your data, e.g. daily.
14. In the Overview, you can schedule this job to run at a suitable interval.

Schedule Record Calculation Task

The steps for creating the Record calculation plan are almost the same as for Plan calculation, the only difference is in step 9 where the Calculation Type should be set to Record Calculation Task.

Assign Access to Users

The package comes with three predefined business roles which speed up configuration. You can adjust their default configuration and assign them to the users who have access to the system.

A detailed description of role definitions and their meaning is described in [Product Info \(SC\) | Business roles](#).

1. From the **Administration** menu go to the **Access Admin > Business Roles Admin** section.
2. Check if all roles with the '[SC]' prefix correspond with your expectations for minimal access and adjust them to your use cases.
3. Assign user accounts to their business roles from the Users tab of each role.

At this point, you completed the installation. Now, you should configure the package for specific business needs and get familiar with its use. The last thing to consider is the integration with a system that processes output compensations.

 Now you can proceed with [Configuration \(Sales Compensation\)](#).

Configuration (Sales Compensation)

This tutorial will guide you through the configuration steps. After you have completed the installation, you can start with the configuration of the Sales Compensation (SC) Accelerator. You can customize the default configuration, extend types or remove those that are not needed.

In this section:

- [Global Package Configuration](#)
- [Apply Customer/Product Filter](#)
- [Customize Plan Header Types](#)
- [Add/Customize Condition Types](#)
- [Define Seller Information in Compensation Header](#)
- [Define Adjustment Wizard Transaction Field in Adjustment Form](#)
- [Define Approval Workflows](#)
- [Turn on/off Forecast Functionality](#)
- [Turn on/off Holdback Functionality](#)
- [Ensure Correct Seller Master Data Deployment](#)

Global Package Configuration

The global configuration data for the package are stored in an element of Advanced Configuration Options. Many of those parameters can be overridden on the Condition Type.

1. From the **Administration** menu go to the System **Configuration** section.
2. Switch to **Advanced Configuration Options**.
3. Search for the `SC_AdvancedConfiguration` element where you can adjust the parameters listed below.
4. Adjust configuration keys according to your specifications.

A list of all parameters, their descriptions, possible values, and other notes are in [Components \(SC\) | Advanced Configuration Options](#).

Sample of '`SC_AdvancedConfiguration`':

```
{
  "sourceType": "DM",
  "sourceName": "SC_TransactionData",
```

```

"pricingDateFieldName": "PricingDate",
"customerIDFieldName": "CustomerId",
"productIDFieldName": "ProductId",
"productNameFieldName": "ProductName",
"baselineFieldName": "InvoicePrice",
"invoicePriceFieldName": "InvoicePrice",
"quantityFieldName": "Quantity",
"marginFieldName": "GrossMargin",
"sellerIDFieldName": "SellerId",
"currencyFieldName": "Currency",
"payoutDays": 2,
"paymentPeriod": "Quarterly",
"quotaFor": "Plan duration",
"depositScheme": "Non-Cumulative"
}

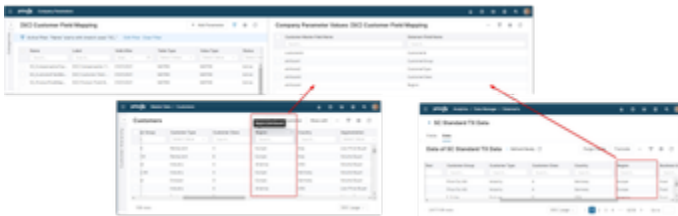
```

Apply Customer/Product Filter

To be able to apply a Customer filter input and Product filter input in a compensation plan, you need to update the mapping fields between Product/Customer master and Datamart fields.

Based on the Datamart structure and Product/Customer master in your partition, you need to update or add more mapping in two Company Parameters:

- SC_CustomerFieldMapping
- SC_ProductFieldMapping



Customize Plan Header Types

How to Customize Your Types

1. Go to **Sales Compensations > Compensation Types**.
2. Select the line you want to edit.
3. Click the Edit button to open the customization dialogue.
4. Adjust parameters according to your needs by replacing default values with custom logic.
5. Click Save Changes to save and confirm changes.

A list of all parameters, their descriptions, possible values, and other notes are in [Libraries \(Sales Compensation\)](#).

Sample of the 'Plan Header Type' which is deployed with the package:

```

Name (uniqueName): Individual Compensation
Header Logic (headerFormulaName): Sales Compensation Header

```

```
Compensation Plan Workflow (workflowFormulaName): Sales Compensation  
Plan  
Compensation Record Workflow (scRecordWorkflowFormulaName):  
Compensation Record
```


Add/Customize Condition Types

This chapter describes procedures for manipulation with Lineltem elements which are primary blocks of calculation for compensations inside the individual plan.

How to Customize Your Types

1. Go to **Sales Compensations > Condition Types**.
2. Look for items with the 'Sales Compensation' in the 'Pricing Logic (formulaName)' column.
3. Select the line you want to edit.
4. Click the Edit button to open the customization dialogue.
5. Adjust parameters according to your needs by replacing default values with custom ones.
6. Click Save Changes to confirm changes.

How to Add a New Type

1. Go to **Sales Compensations > Condition Types**.
2. Click 'Add Condition Type' and a configuration dialogue will open.
3. Fill in fields you want to have specifically defined in the condition type.
 -  Be aware, parameters can inherit default values from Advanced Configuration Options. Empty field means the parameter inherits a value.
 - "Condition Type Name" shows a list of Sales Compensation. Options compatible with this package are listed in [Type Overview \(SC\) | Condition Types](#).
4. Click Add to save your new type.

A list of all parameters, their descriptions, inheritance, possible values, and other notes are in section [Components \(SC\) | Condition Type Attributes](#).

Define Seller Information in Compensation Header

Add a new read-only input which will show the detailed information about selected Seller input on the Compensation Plan Header.

How to Customize Your Seller Information Display

1. Go to Company Parameters.
2. Go to Pricefx Accelerators parent folder Sales Compensation folder.
3. Select SC_HeaderSellerDetailsMapping Company Parameter name.
4. Add a new record or adjust the existing one to display which the seller information you want.

Sample of SC_HeaderSellerDetailsMapping which is deployed with the package:

Company Parameter Values: Header Seller Details Mapping



<input type="checkbox"/> Field Label	Seller Field
<input type="text" value="Search..."/>	<input type="text" value="Search..."/>
<input type="checkbox"/> Seller ID1	sellerId
<input type="checkbox"/> Seller Name	name
<input type="checkbox"/> Reports To	reportsTo
<input type="checkbox"/> First Name1	attribute1
<input type="checkbox"/> Last Name	attribute2
<input type="checkbox"/> Termination Date	attribute5

Define Adjustment Wizard Transaction Field in Adjustment Form

You can display any transaction field in the Adjustment Form (Adjustments under the Sales Compensation module).

How to Customize Your Adjustment Wizard Transaction Field Display

1. Go to Company Parameters.
2. Go to Pricefx Accelerators parent folder Sales Compensation folder.
3. Select SC_AdjustmentWizardTransactionFieldMapping Company Parameter name.
4. Add a new record or adjust the existing one to display which the Adjustment Field Name you want in the Adjustment Form.

Sample of SC_AdjustmentWizardTransactionFieldMapping which is deployed with the package:

Company Parameter Values: SC Adjustment Wizard Transaction Field Mapping

<input type="checkbox"/> Adjustment Field Name	Datamart Field Name
<input type="text" value="Search..."/>	<input type="text" value="Search..."/>
<input type="checkbox"/> Customer ID	-ADJUST_ME-
<input type="checkbox"/> Invoice Price	-ADJUST_ME-
<input type="checkbox"/> Margin	-ADJUST_ME-
<input type="checkbox"/> Pricing Date	-ADJUST_ME-
<input type="checkbox"/> Product ID	-ADJUST_ME-
<input type="checkbox"/> Quantity	-ADJUST_ME-
<input type="checkbox"/> Seller ID	-ADJUST_ME-

Define Approval Workflows

Two types of approval workflows can be configured to support business conditions and flow inside an organization. The installation package already deployed and configured workflow types (SCA and SCR, Adjustment) in **Administration > Logics > Workflow Logics > General Workflow Logics**. You just need to convert your business rules to workflow parameters in Company Parameters.

Compensation Plan Approval Workflow

In this step, you should configure workflow conditions for the Compensation Plan. You will follow detailed steps from the manual for the [Accelerate Approval Workflow Package](#) which is part of this package.

1. Go to **Company Parameters** in the main menu.
 2. Locate PFXTemplate > Approval Workflow.
 3. Define workflow steps in the 'ApprovalWorkflowsSetup' parameter.
 - 'Workflow Type' must be **SCA**.
 4. Define conditions in the 'ApprovalConditions' parameter.
 - The list of variables is on the [Workflow Variables \(SC\) | Compensation Agreement approval workflows](#) page.
 5. Define approvals in the 'Approvers' parameter.
 6. Test your workflow.
- i** You don't need to submit a plan to test it. Just recalculate and verify that the workflow is prepared to be triggered.

Example for a two-step workflow for Compensation Plan with conditions and approvals:

The screenshots show the configuration of a two-step approval workflow for a Compensation Plan. The interface is divided into a left sidebar with categories and a main content area with two panels: 'Company Parameter Values' and 'Company Parameter Values: [Parameter Name]'. The 'Approval Workflow Setup' panel shows a table with columns for Workflow Type, Step ID, Step Order, Step Label, and Reason. The 'Approvers at each step' panel shows a table with columns for Name, Label, Valid After, Table Type, and Value Type. The 'Approval conditions at each step' panel shows a table with columns for Name, Label, Valid After, Table Type, Condition ID, Condition Description, Condition, and Skip.

Workflow Type	Step ID	Step Order	Step Label	Reason
SCA	1	1	Senior Sales Management	Management approval needed
SCA	2	2	VP of Sales	VP level needed
SCR	1	1	SalesOps	Compensation too high

Workflow Type	Step ID	Approver ID	Approver Type	Approver Name	Skip
SCA	1	1	BUSINESS ROLE	DEMO_SC_SSM	No
SCA	2	2	USER	john.doe	No
SCR	1	1	BUSINESS ROLE	DEMO_SC_SSM	No

Workflow Type	Step ID	Condition ID	Condition Description	Condition	Skip
SCA	1	1	Previous Compensation 10 000+	"line.PreviousCompensation" > 10000	No
SCA	2	1	Compensation is more than 15 %	"line.Compensation %" > 15.0	No
SCR	1	1	Compensation is more than 90	attribute1? > 90.0	No

Compensation Record Approval Workflow

This workflow type configuration is identical to the one above. It uses **SCR** as the 'Workflow Type'.

An example from the previous step shows one step workflow configuration. The list of variables is on the [Workflow Variables \(SC\) | Agreement Record approval workflows](#) page.

i To check the approvals for Compensation Records, you must approve the Compensation Plan first.

Adjustment Custom Form Workflow

This workflow type configuration is identical to the one above. It uses **Adjustment** as the 'Workflow Type'. An example from the previous step shows one step workflow configuration.

In the current development, you only need to define the approval condition as empty to display the approval step:

Workflow Type	Step ID	Cond.	Condition Description	Condition	Step	Attribute 4	Attribute 5	Attribute 6
	Adjustment		Check Adjustment	No				

Turn on/off Forecast Functionality

Usage of forecasting on Sales Compensation Plans is allowed by default but if needed, you can switch it off just by updating the Company Parameters table.

1. Go to **Company Parameters** in the main menu.
2. Find the SC_FeatureConfig table.
3. For the FORECAST Feature Name change the Status value to OFF.
4. Inputs related to forecasts will no longer appear in the plan header or anywhere in the system.

Turn on/off Holdback Functionality

Holdbacks define how much of the compensation should be held back from the payee and paid only in the last period of the plan. Holdbacks are enabled by default but if needed, you can switch them off just by updating the Company Parameters table.

1. Go to **Company Parameters** in the main menu.
2. Find the SC_FeatureConfig table.
3. For the HOLDBACK Feature Name change the Status value to OFF.

Ensure Correct Seller Master Data Deployment

When this accelerator is deployed via PlatformManager, the data for PlatformManager is taken from *data/SellerAttributes* and the following specifics are of significance.

In the accelerator, the *SellerAttribute* folder is packaged within the *definition* folder. It means its content is deployed during the "deploy-logic" step. This step overrides all changes made by the "setup-seller-master" step.


To make sure, all changes in Seller Master data stay in place, exclude the *SellerAttribute* folder from the *definition* folder (but leave it in the *data* folder). Then list the folder in the "excludes" section in the pom.xml file. This way it will be added into the *data* folder, but it will be excluded from the *definition* folder.

Advanced Configuration (Sales Compensation)

The main place for package configuration is an Advanced Configuration option named [SC_AdvancedConfiguration](#). Option values are configured mostly during a package deployment process (refer to [installation](#)) and can be adjusted later here. Some options can be overridden on a Condition Type or Line Item level where the lower level has a priority.

The following table shows a list of all configuration parameters stored in SC_AdvancedConfiguration:

Name	Description	Values	Condition Type can be overwritten?
sourceType	Type of source data.	Select one of these options: <ul style="list-style-type: none"> DM - Datamart DS - Data Source 	✓
sourceName	Name of the source used to query and allocate the compensation value.		✓
pricingDateFieldName	Contains a pricing date in the Source Name.		✓
customerIDFieldName	Contains a customer ID in the Source Name.		✓
productIDFieldName	Contains a product ID in the Source Name.		✓
productNameFieldName	Contains a product name. It is used for the <i>Transaction List</i> report only.		
baselineFieldName	The field name of a Value Base field is used to calculate the base value. In most cases, it will be a revenue or margin field; in some cases it can be volume. This depends on the used Condition Type.		✓
invoicePriceFieldName	Contains an Invoice Price in the Source Name. It is used for the Adjustments		✓
quantityFieldName	Contains a Quantity in the Source Name. It is used for the Adjustments		✓
marginFieldName	Contains a Margin in the Source Name. It is used for the Adjustments		✓
sellerIDFieldName	Contains an identification of a sales agent.		✓
currencyFieldName	Contains the main Source Name (DM or DS) currency.		
payoutDays	Number of days after the end date of the payment period when the payout happens.		✓
paymentPeriod	Sets a frequency of payouts. The user can specify it also on a Line Item level.	Select one of these options: <ul style="list-style-type: none"> Monthly Quarterly Semi-Annually Annually 	✓
quotaFor		Select one of these options:	✓

	The target defined on the line item is evaluated for each payment period (see <i>paymentPeriod</i>) or for the whole plan duration.	<ul style="list-style-type: none"> • Payment Period - Gets a base value for every period, then compares and calculates the compensation. • Plan Duration - Gets a base value for the whole plan duration, then compares and calculates the compensation. 	
depositScheme	Parameter providing guidance on compensation accumulation across payment periods.	<p>If "quotaFor" is "Plan duration", you need to define how to calculate the compensation value:</p> <ul style="list-style-type: none"> • Non-Cumulative - The compensation value is calculated for the current period based on a cumulation of the base value, then divided by the number of periods (12 months or 4 quarters, 2 for semi-annually, 1 year). • Cumulative - The same as with calculation of Non-Cumulative above, but the compensation value of the previous period is excluded. 	

Example of SC_AdvancedConfiguration:

```

{
  "sourceType": "DM",
  "sourceName": "SC_TransactionData",
  "pricingDateFieldName": "PricingDate",
  "customerIDFieldName": "CustomerId",
  "productIDFieldName": "ProductId",
  "productNameFieldName": "ProductName",
  "baselineFieldName": "InvoicePrice",
  "invoicePriceFieldName": "InvoicePrice",
  "quantityFieldName": "Quantity",
  "marginFieldName": "GrossMargin",
  "sellerIDFieldName": "SellerId",
  "currencyFieldName": "Currency",
  "payoutDays": 2,
  "paymentPeriod": "Quarterly",
  "quotaFor": "Plan duration",
  "depositScheme": "Non-Cumulative"
}

```

Explanation of selected parts of the configuration above:

Example 1:

```
"payoutDays": 2,  
"paymentPeriod": "Quarterly",  
"quotaFor": "Payment Period",  
"depositScheme": ""
```

The layout on the line item of a Compensation Plan (see details in [Create Compensation Plan \(Sales Compensation\)](#)) based on the configuration:



Flat Bonus

Input Parameters

Product(s)

Product Class: A



Quota *

100

Target Incentive *

15

Payment Period *

Quarterly



Quota For

Payment Period

Deposit Scheme

Baseline Field ?

Invoice Price

- Quota input (100) and Target Incentive (15) are understood as per Payment Period input (in this case, it is Quarterly by default, you can select another input such as Monthly, Semi-Annually, or Annually).
- Then, you compare those inputs with your baseline value in the Source Name, Source Type (e.g. SC_TransactionData DM) for each payment period (Q1, Q2, Q3, Q4) and calculate the compensation based on the condition type you selected.
- End of each payment period (Q1, Q2, Q3, Q4), the payout date is planned to pay for the sales agent at the second day (2) later, excluding the weekends.

Example 2:

```
"payoutDays": 4,  
"paymentPeriod": "Quarterly",  
"quotaFor": "Plan duration",  
"depositScheme": "Non-Cumulative"
```

Flat Bonus

Input Parameters

Customer(s)

Select Customer(s) ▼



Product(s)

Product Class: A



Quota *

130

Target Incentive *

20

Payment Period *

Quarterly ▼

Quota For

Annual

Deposit Scheme

Non-Cumulative

- Quota For is Plan Duration. Quota input (130) and Target Incentive (20) are understood as a whole compensation plan duration (it should be the Start Date - End Date input on the Compensation Plan).” (see details in [Create Compensation Plan \(Sales Compensation\)](#)).
- You compare those inputs with your cumulated baseline value in the Source Name, Source Type (e.g. SC_TransactionData DM) for payment period (Q1, Q2, Q3, Q4).
- For calculation details for this case see <https://pricefx.atlassian.net/wiki/spaces/ACCDEV/pages/3807543836/Type+Overview+SC#Agreement-Records-Calculation>.
- End of each payment period (Q1, Q2, Q3, Q4), the payout date is planned to pay for the sales agent at the fourth day (4) later, excluding the weekends.

Summary

There are 3 different calculation ways:

1. Quota For: **Payment Period**
2. Quota For: **Plan Duration** - Deposit Scheme: **Non-Cumulative**
3. Quota For: **Plan Duration** - Deposit Scheme: **Cumulative**

Upgrade (Sales Compensation)

This tutorial will guide you through the upgrade of the Sales Compensation Accelerator.

In this section:

- [Prerequisites](#)
- [Upgrade Steps](#)

Prerequisites

Before you start, ensure that you have:

- Access to a partition on the Pricefx server (8.0 or newer). You will need:
 - Server URL
 - Partition name
 - Username and password for a partition user with sufficient rights for using the Accelerator
- Access to Pricefx PlatformManager
 - Username and password for PlatformManager user
 - The user must have the following permissions for your partition (to which you plan to deploy the Accelerator):
 - Permission *Marketplace Templates - deploy*
- Familiarize yourself with the steps required after the upgrade - [Manual Upgrade Steps \(SC\)](#).
- Optionally, read about changes in [Release Notes \(SC\)](#).

Upgrade Steps

Select Partition for Deployment and upgrade logics.

1. In PlatformManager, navigate to **Marketplace** > **Accelerator Packages**, find the *Sales Compensation - Upgrade*.

2. Click **Deploy** and select a partition to which you want to upgrade.
i Only logics are deployed. The configuration remains without changes.
3. Click **Deploy**.
4. A warning dialogue will appear. After you read the warning text and you agree with the conditions, you can click **Continue**.
5. The first part of the upgrade is complete. Go to your partition and continue with the manual steps required after the upgrade.
The exact process depends on the original version of the package, you will find all details in [Manual Upgrade Steps \(SC\)](#).

Troubleshooting (Sales Compensation)

In this section, you can find troubleshooting tips that will help you during the support procedure.

Where do I find a package version?

For efficient communication with the support team, you should know which package version is used.

1. Go to **Company Parameters**.
2. Search for the 'deployedAccelerators' parameter.
3. Find the key 'sales-compensation', its value contains 'templateVersion'.
4. Version is a value stored in 'templateVersion'.

Known Limitations (Sales Compensation)

Adjustments Limits [Solved in 1.6.0]

In versions 1.4.0, 1.5.0, when you submit the Adjustment, with filters on Transactions data returning data exceeding 25 000 rows, you will often get timeout. This issue was resolved by adding support for mass addOrUpdate for COARs (see PFUN-17298 in the [1.6.0 release notes](#)).

Warnings in Compensation Plan

In versions 1.4.0, 1.5.0, 1.5.1, 1.6.0, warnings on the Compensation Plan level are not displayed properly. There is a product enhancement request which is approved and scheduled to be tackled in 2023:

\$ [PM-2369](#) - (BE) api.addWarning doesn't work in Sales Compensation and Rebate Manager **RELEASED** .

These warnings should be displayed if:

- Any of the sellers selected in the Seller input are terminated.
- Any of the sellers selected in the Seller input have a missing salary.
- Any of the sellers selected in the Team input have been skipped because of invalid validity dates.
- Any of the sellers selected in the Team input have been skipped because validity dates contradict planned start/end date settings.

Technical User Reference (Sales Compensation)

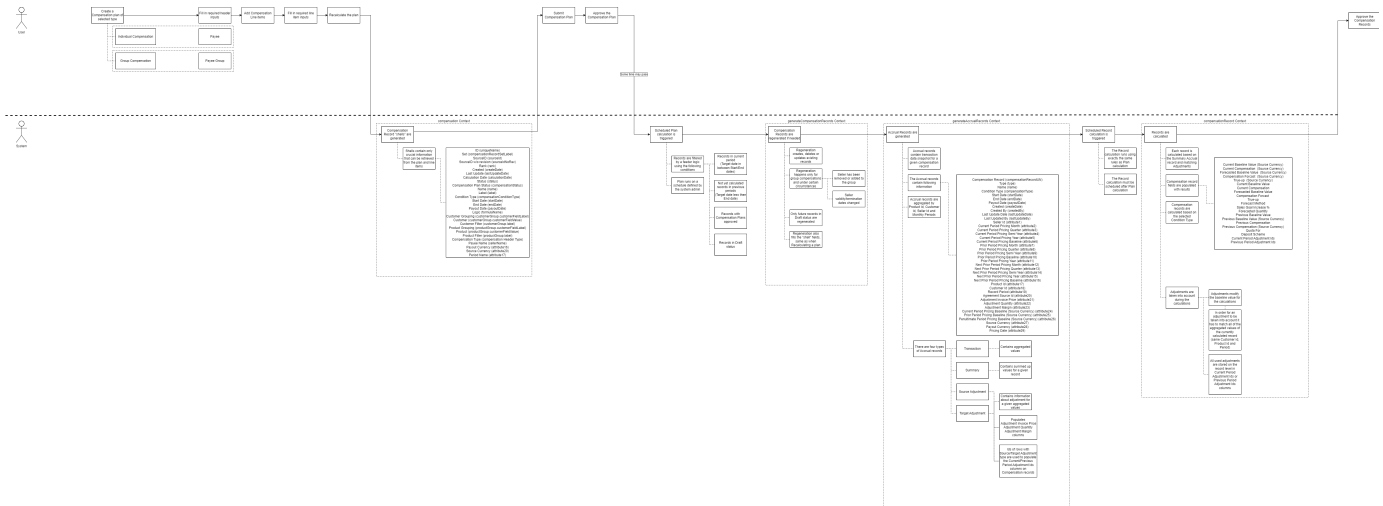
- [Flow Diagrams \(Sales Compensation\)](#)
- [Logic Documentation \(Sales Compensation\)](#)

- Company Parameters (Sales Compensation)
- Advanced Configuration Options (Sales Compensation)
- System Objects Attributes (Sales Compensation)
- Business Roles and Their Permissions (Sales Compensation)
- Compensation Types (Sales Compensation)
- Condition Types & Calculation Examples (Sales Compensation)
- Adjustments (Sales Compensation)
- Holdback (Sales Compensation)
- Approval Workflow (Sales Compensation)

Flow Diagrams (Sales Compensation)

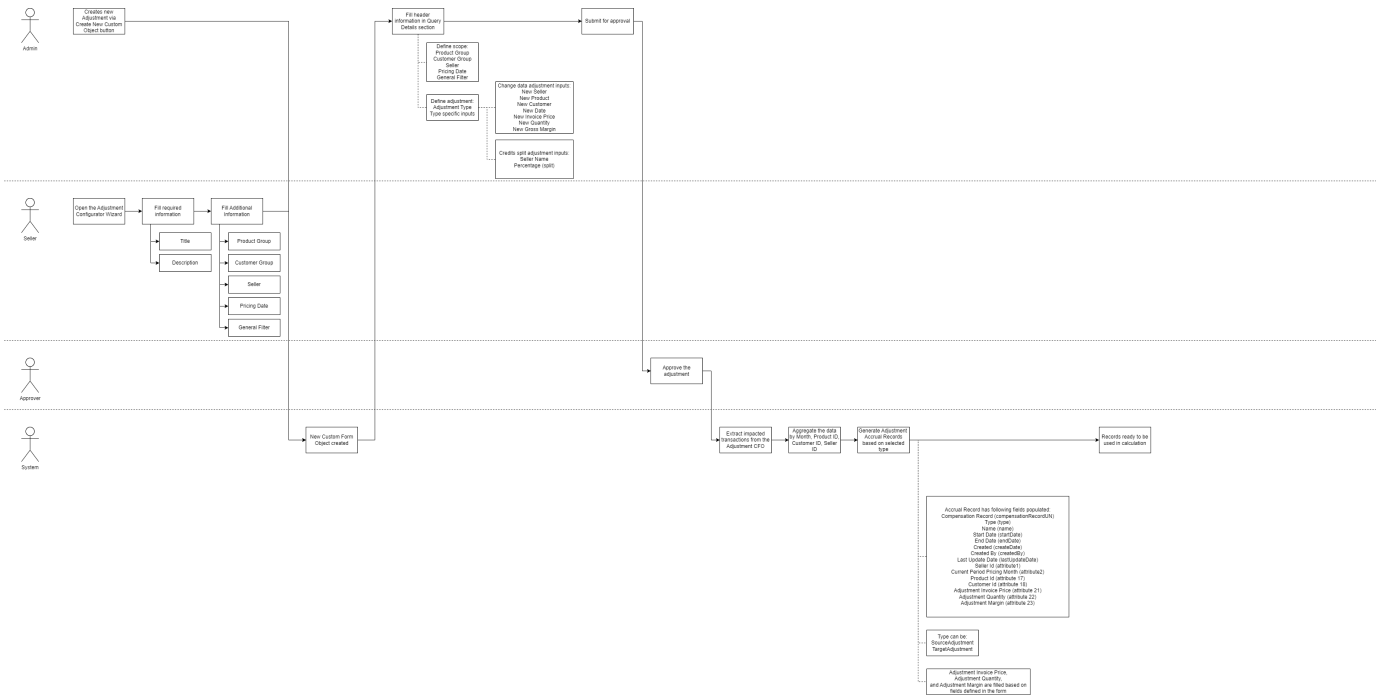
Calculation Flow

The following diagram represents the main calculation flow of the solution, starting from the creation of a plan, through generation of accrual records, adjustments and ending with the calculation results. This is the core of the solution.



Adjustments Flow

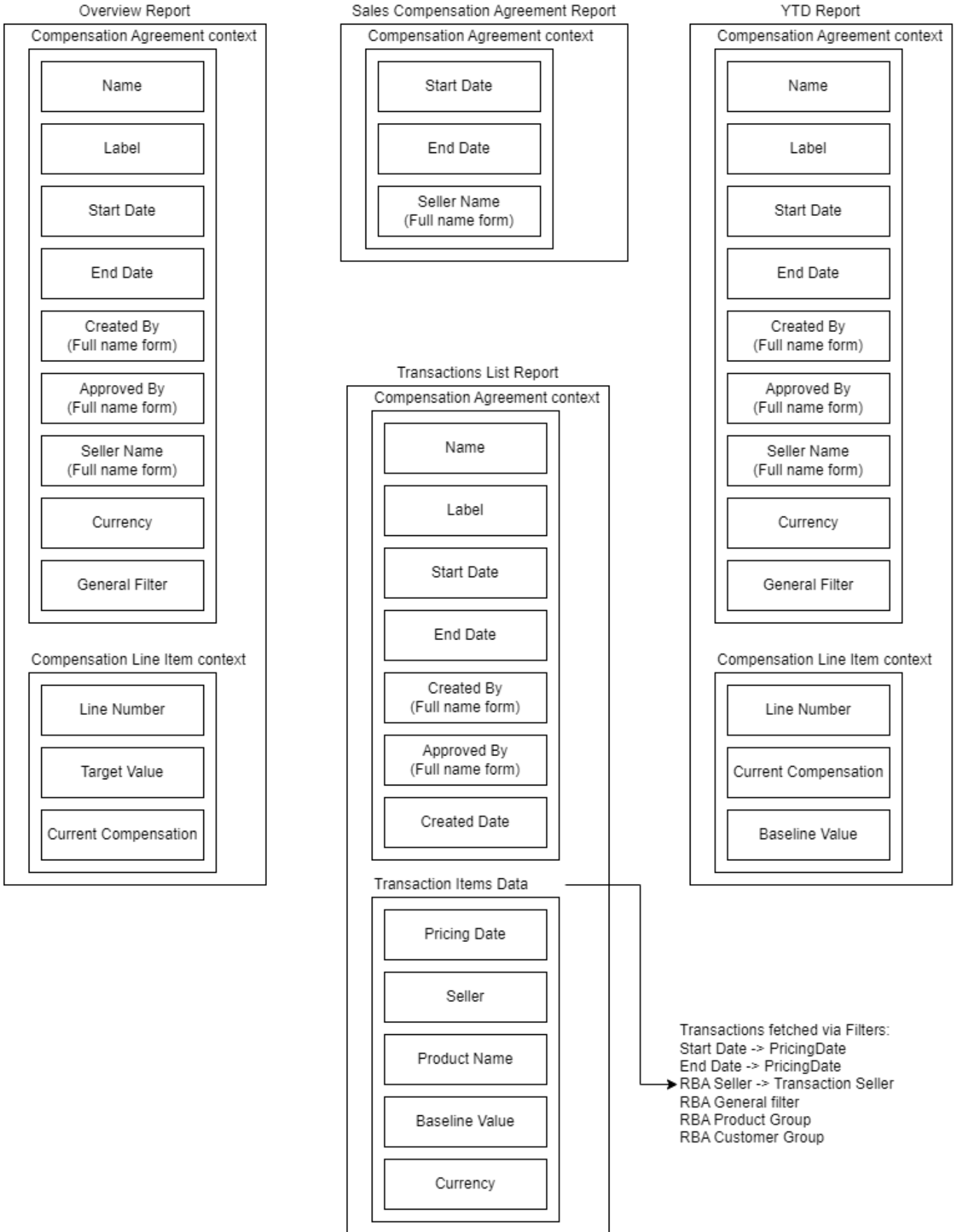
The following diagram represents the flow of adjustments, how they are created, what impact they have and what the result of an adjustment creation is.



Flow Diagrams 2 (Sales Compensation)

Simplified Data Flow: Compensation Plan Compensation Record

Reports Data Flow



Logic Documentation (Sales Compensation)

- SC_CompensationDashboardsLib
- SC_CompensationProcessingLib
- SC_CompensationTypesLib
- SC_CompensationHeader
- SC_CompensationHeaderSellerConfigurator
- SC_Forecast_Configurator
- SC_HoldBack_Configurator
- SC_Compensation
- SC_Dashboard_SellerOverview_Configurator
- SC_Dashboard_SellerOverview
- SC_Dashboard_SellerOverview_CompensationReferenceTable_PlanDetails_ContextLink
- SC_Dashboard_SellerOverview_CompensationReferenceTable_TransactionDetails_ContextLink
- SC_Dashboard_TransactionsAndStatistics_Configurator
- SC_Dashboard_TransactionAndStatistics
- SC_Dashboard_AdminYTDSummary_Configurator
- SC_Dashboard_AdminYTDSummary
- SC_Dashboard_AdminYTDSummary_PayoutsOverviewTable_PayoutDateDetails_ContextLink
- SC_Dashboard_AdminYTDSummary_PayoutsOverviewTable_PayoutSellerDetails_ContextLink
- SC_Dashboard_AdminYTDSummary_PayoutsOverviewTable_SellerYearlyOverview_ContextLink
- SC_Dashboard_PayoutsAndPlans_Configurator
- SC_Dashboard_PayoutsAndPlans
- SC_Dashboard_PayoutsAndPlans_PayoutsTable_Details_ContextLink
- SC_EmbeddedCompensationDetailDashboard
- SC_Compensation_RecordCalculationFeeder
- SC_AdjustmentHeader
- SC_AdjustmentHeaderConfigurator
- SC_AdjustmentWizardExecutor
- SC_AdjustmentWizardInputs
- SC_QuotingPluginLib
- SC_NoConditionTypeFilter
- SC_SPIFFConditionTypeFilter
- SC_DefaultConditionTypeFilter

SC_CompensationDashboardsLib

This logic contains common utility functions that are used by the Sales Compensation package. The library focuses on dashboards and operations related to them - mostly data fetching and processing.

See also the [architecture documentation](#).

Elements Description

- **ConstConfig** - Contains definitions of global constant variables that are used throughout the logic. Any configuration-like constants should be stored here to avoid hard-coding and to allow for better maintainability of the solution.
- **ConfigurationUtils** - Extension of the common element from SC_CompensationDashboardsLib, contains SC package specific configuration structures and loading operations. This is the place where all Advanced Configuration setup fetching and storage takes place.

- **CompensationTypeUtils** - Contains methods related to fetching and accessing data from the Compensation Type object (COCT).
- **CompensationRecordUtils** - Contains methods related to fetching and accessing data from the Compensation Record object (COR). This util also defines how the Compensation Record data are structured.
- **CompensationAgreementUtils** - Contains methods related to fetching data from the Compensation Plan object (CO).
- **CompensationAgreementLineItemUtils** - Defines methods for accessing and building the structures for Compensation Plan Line Items ("COLI" currently). The data is extracted from the already fetched and processed Compensation Plan objects.
- **ConfiguratorUtils** - Contains methods used in various configurators for dashboards. These are mostly input generation methods.
- **DateUtils** - Contains methods related to date in any way, such as filter generation for years or start /end dates.
- **InputUtils** - Contains methods related to common user dashboard inputs.
- **InputValidationUtils** - Contains methods used for input validation in dashboards - such as checking if input is not empty or has valid values.
- **FormatUtils** - Contains methods used for formatting output values to appropriate money, percentage etc. formats. Currency symbol generation methods are also stored here.
- **PortletUtils** - Contains methods that generate ready to use portlets, such as summary portlets.
- **QueryUtils** - Contains methods related to any type of Datamart / Data Source lookups and queries.
- **CurrencyUtils** - Contains methods related to currency (formats, fetching or label generation).
- **PaybackTypeUtils** - Contains methods related to fetching and processing system data for various types of compensation types.
- **PaybackAgreementUtils** - Contains everything related to compensation plans and also contains methods for generation of data structures as well as for fetching the data and displaying it.
- **UserUtils** - Contains methods related to users, such as checking if a user belongs to a given user group.
- **SellerUtils** - Contains methods related to Seller master data, such as filtering based on user groups.
- **ContextLinkUtils** - Contains methods related to Context Link functionality present in the solution (e.g. forwarding from one dashboard to another).

SC_CompensationProcessingLib

This logic contains common utility functions that are used by the Sales Compensation package. The library focuses on data processing related to the core of this package - that means calculation of plans and records mostly.

See also the [architecture documentation](#).

Elements Description

- **ConstConfig** - Contains definitions of global constant variables that are used throughout the logic. Any configuration-like constants should be stored here to avoid hard-coding and to allow for better maintainability of the solution.
- **ConfigManager** - Strictly defined structure that allows management of solution configuration. The benefit is that the whole configuration is contained within one "class", it cannot be modified and is cached.
- **CompensationInputParameter** - "Class" like element that provides a creation method for a structure called InputParameter which stores all user inputs for further processing or access.
- **InputUtils** - Extension of SC_CompensationProcessingLib InputUtils element with Sales Compensation related methods. Contains mostly methods for creating inputs on the line item level.
- **InputValidationUtils** - Extension of SC_CompensationProcessingLib InputValidationUtils, contains checks for Sales Compensation specific input types and their validation.

- **QueryUtils** - Contains methods related to Datamart queries, mostly for transaction type data.
- **FormatUtils** - Extension of SC_CompensationProcessingLib FormatUtils, extends the formatting by coloring options.
- **CalculationUtils** - Contains methods for final steps of the core logic, creation of Compensation Records and enrichment of Compensation Record data based on the calculation base.
- **CacheDataManagementUtils** - Contains methods for various caching purposes.
- **CompensationDataCalculator** - Contains methods related to main calculations, that is for example calculation of the current and previous compensation in various contexts. Calculations related to transaction data can also be found here.
- **CompensationAgreementUtils** - Contains methods that focus on Compensation Agreement objects as well as fetching and processing data coming from those objects.
- **CompensationPlanLineItemUtils** - Contains methods that allow fetching the Plan line items.
- **CompensationRecordUtils** - Contains methods that define a way of fetching the CompensationRecords as well as provide a way of building filters for the fetch based on the provided data.
- **ConditionTypeUtils** - Contains methods that allow fetching the Condition Types.
- **CompensationAgreementUtils** - Extension of SC_CompensationProcessingLib PaybackAgreementUtils with Sales Compensation related methods. The main addition is the focus on Compensation Agreement objects as well as fetching and processing data coming from those objects.
- **CompensationRecordUtils** - Extension of SC_CompensationProcessingLib PaybackRecordUtils with Sales Compensation related methods. Defines a way of fetching the CompensationRecords as well as provides a way of building filters for the fetch based on the provided data.
- **BasicCompensationValuesDataUtils** - Helper util element that supports compensation calculations, summaries etc.
- **CompensationCalculationParameter** - "Class" like element that provides a creation method for a structure called CalculationParameter which stores all calculation related closures for further processing or access.
- **CompensationValueCalculator** - Contains methods that describe the calculation methods for various types of compensations: target, stepped, growth or multi.
- **FilterUtils** - Extension of SC_CompensationProcessingLib FilterUtils with Sales Compensation related methods. Mostly related to building a product/customer filter structure.
- **ReportUtils** - Contains utils that are used in all reports in the package. This util includes data fetches for reports as well as generation of report specific labels.
- **SellerUtils** - Contains methods that are used for all Seller related data manipulation - that is fetching, retrieval based on some specific fields etc.
- **AdjustmentUtils** - Contains methods that are related to Adjustment Custom Form, their creation, processing and access of data.
- **AdjustmentConfiguratorUtils** - Contains methods used in Adjustment related configurators (Adjustment Form and Adjustment Creation Wizard) and mainly consists of input generation methods.
- **AdjustmentValidationUtils** - Contains methods related to validation of Adjustment inputs.
- **CompensationTypeUtils** - Contains methods related to CompensationTypes. Provides methods for easy access to all necessary information of a given CompensationType, as well as processing methods for its data such as reading Target or Compensation values from line items.
- **AccrualForecastUtils** - Contains calculation methods related to Forecasts.
- **ChartUtils** - Contains methods that generate charts. These charts are displayed in the outputs of the Compensation Plan, hence they have been separated from the SC_CompensationDashboardsLib.
- **DateUtils** - Contains methods related mostly to date and period definition. Reuses and extends a lot of methods from SharedLib.
- **PeriodDataUtils** - Related to DateUtils, this util handles the Period structure creation and data access to it.
- **MathUtils** - Contains math related methods, such as percentage calculations.

SC_CompensationTypesLib

The library stores all types of currently supported Compensation Types that can be added as line items in Compensation Agreements. Each type has a strictly defined structure: Calculation Type, Target Type, Target Value Type, Compensation Value Type as well as a way to calculate the compensation. The element names match 1-1 the Condition Types defined in the system and should be carefully maintained if any changes arise.

See also the [architecture documentation](#).

Elements Description

- SingleTargetAmount
- SingleTargetPercent
- MultiTargetAmount
- MultiTargetPercent
- SteppedAmount
- SteppedPercent
- GrowthAbsoluteAmount
- GrowthAbsolutePercent
- GrowthPercentAmount
- GrowthPercentPercent
- FlatBonus
- MultiTargetBonus
- SteppedBonus
- VariablePayLinearAmount
- VariablePayMultiTargetAmount
- VariablePaySteppedAmount
- ZeroQuotaPercent
- RepetitiveQuotaAmount

SC_CompensationHeader

This is currently the only header logic that defines the header behavior of Compensation Plan. The logic defines the inputs and outputs from the header and further operations that happen in the main core logic.

See also the [architecture documentation](#).

Elements Description

- **ConfigManager** - Initialization of a ConfigManager structure defined in SC_CompensationProcessingLib.
- **HeaderUtils** - Contains methods for creating header inputs and outputs, main codebase of the logic. Calculation methods for Previous and Current compensation can also be found here.
- **SyntaxCheckAbort**
- **HeaderInputs** - Contains initialization of all header inputs using HeaderUtils (even the hidden ones, such as the type of the header).
- **HeaderOutputs** - Adds outputs to the header using HeaderUtils.

SC_CompensationHeaderSellerConfigurator

This is the configurator for Seller selection on the header level. It is used only in header types that contain a singular Seller. It displays additional information that is configured to be displayed in SC_HeaderSellerDetailsMapping Company Parameter table.

See also the [architecture documentation](#).

Elements Description

- **SellerInput** - Returns a user input that allows a Seller selection.
- **SellerDetailsInput** - Returns a read-only user input that, based on selection in SellerInput, displays additional Seller information provided in the SC_HeaderSellerDetailsMapping Company Parameter table.

SC_Forecast_Configurator

This is the configurator for the Forecast input on the header level. It is used to display forecast inputs for the user.

See also the [architecture documentation](#).

Elements Description

- **ForecastMethod** - Generates the Forecast Method options selection input that allows the user to provide forecast method.
- **SalesGoalIncreasePercent** - Presents sales goal increase % value user input.
- **PreviousForecastMethod** - Stores the old value of ForecastMethod. It is used to check the current value of ForecastMethod to reset value for SalesGoalIncreasePercent.

SC_HoldBack_Configurator

This is the configurator for the Holdback input on the header level. It is used to display holdback inputs for the user that allow to select a holdback value and type.

Elements Description

- **HoldbackInputs** - Displays two inputs that allow the user to set the holdback value and holdback type.

SC_Compensation

This is the core logic for the whole package. The logic works in four contexts:

1. *Agreement* context (line items) which as a result outputs a set of Compensation Records with some data prefilled.
2. *Generate Compensation Record* context which regenerates records if any of the main source data changed.
3. *Generate Accrual Records* context which constructs the snapshot of the current transaction data and stores them in Accrual Records to be used in calculations later on.
4. *Compensation Record* which does the actual calculation that is run via the Sales Compensation Calculation object on a set schedule.

This logic - due to its size - has "context separators" which make it easier to see which elements run in which context. The separators are empty elements with the following name pattern: `___SOME_NAME_CONTEXT1_CONTEXT2___`

- `SOME_NAME` describes what is in the section.
- `CONTEXT1, CONTEXT2` define the contexts this section runs in.

See also the [architecture documentation](#).

Elements Description

- **CompensationConfig** - Initialization of a ConfigManager structure defined in SC_CompensationProcessingLib.
- **CompensationTypeCode** - Retrieves the type name of the currently processed Compensation Line Item. This is the Name column value from ConditionTypes. This code is used to access the appropriate element from the CompensationTypesLib later on.
- **CalculationType** - Defines the calculation type of a currently processed Compensation Type. Can be one of these values: Conditional, Growth, Stepped.
- **QuotaType** - Defines the quota type of a currently processed Compensation Type. Can be one of these values: None, Single, Multi.
- **QuotaValueType** - Defines the target value type of a currently processed Compensation Type. Can be one of these values: Amount, Percent.
- **CompensationValueType** - Defines the compensation value type of a currently processed Compensation Type. Can be one of these values: Amount, Amount Per Unit, Percent, None.
- **DatamartCurrency** - Defines the currency used by the main Datamart. The Datamart configuration is stored in SC_AdvancedConfiguration.
- **CustomerGroupInput** - User input for Customer Group, used for filtering later on.
- **ProductGroupInput** - User input for Product Group, used for filtering later on.
- **SingleQuotaInput** - Returns a user input for a particular Compensation Type that has the quota defined as Single. Allows input of a quota factor.
- **SingleCompensationInput** - Returns a user input for a particular Compensation Type that has the target defined as Single. Allows input of a compensation value.
- **TargetIncentiveInput** - Returns a user input for a particular Compensation Type that has the calculation type defined as Bonus. Allows input of a target incentive value.
- **VariablePayPercentageInput** - Returns a user input for a particular Compensation Type that has the target defined as Variable Pay or Stepped Variable Pay. Allows input of a variable pay percentage value.
- **MultiQuotaInput** - Returns a user input for a particular Compensation Type that has the quota defined as Multi and Calculation Type as one of Conditional, Growth or Stepped. Allows input of a combination of both quota factor and compensation value in a multitiered fashion.
- **MultiTargetBonusInput** - Returns a user input for a particular Compensation Type that has the quota defined as Multi and Calculation Type as Bonus. Allows input of a combination of both quota factor and bonus payout rate factor in a multitiered fashion.
- **SteppedBonusInput** - Returns a user input for a particular Compensation Type that has the quota defined as Multi and Calculation Type as Stepped Bonus or Stepped Variable Pay. Allows input of a combination of both quota factor and bonus payout rate factor in a multitiered fashion.
- **MultiTargetVariablePayInput** - Returns a user input for a particular Compensation Type that has the quota defined as Multi and Calculation Type as Variable Pay. Allows input of a combination of both quota factor and variable pay in a multitiered fashion.
- **CompensationCapInput** - If the Compensation Caps feature is turned on this input, it allows users to enter the cap value.
- **PaymentPeriodInput** - Displays a used input for a Payment Period selection as an option. Available values are Monthly, Quarterly, Semi-Annually, Annually.
- **GeneralFilterInput** - If the "general filter on line item" feature is turned on, this input displays and allows you to set the General filter on line item level.
- **QuotaForInput** - Displays a read-only user input containing the QuotaFor value stored in Condition Types table. If the value is not present, the value from Advanced Configuration SC_AdvancedConfiguration is taken instead.
- **DepositSchemeInput** - Displays a read-only user input containing the DepositScheme value stored in Condition Types table. If the value is not present, the value from Advanced Configuration SC_AdvancedConfiguration is taken instead.

- **BaselineFieldNameInput** - Displays a read-only user input containing the BaselineFieldName value stored in Condition Types table. If the value is not present, the value from Advanced Configuration SC_AdvancedConfiguration is taken instead.
- **SyntaxCheckAbort**
- **HeaderSellerGroupInput** - Retrieves the Seller group input value from the header level.
- **HeaderCurrencyInput** - Retrieves the currency value from the header level.
- **HeaderTypeInput** - Retrieves the type of the header.
- **ObjectValidation** - Performs validation of the state of a currently processed item in order to assess whether the logic execution should continue.
- **SellerIdsInput** - Returns the Seller group converted into a list of individual Seller IDs.
- **CompensationInputDefinition** - Returns CompensationInputParameter that stores all the inputs taken from user/header etc. This parameter is used later on for other processing/access operations.
- **InputValidation** - Performs validation on all inputs that were provided. If any input is marked as invalid, an exception is thrown with a proper message to the user.
- **PreviousPeriodAdjustmentData** - Returns the previous period adjustment data used later on in calculations.

CurrentPeriodAdjustmentData - Returns the current period adjustment data used later on in calculations.

- **PreviousPeriodData** - Returns a value of the previous period compensation based on the current context.
- **CurrentPeriodData** - Returns a value of the current period compensation based on the current context.
- **BaselineValueFieldLabel** - Returns a field in a Datamart that will be used for baseline calculations. The field is defined in the Datamart Advanced Options configuration.
- **PreviousBaselineValue** - Extracts the baseline value from the previous period data fetched in one of the previous elements.
- **CurrentBaselineValue** - Extracts the baseline value from the current period data fetched in one of the previous elements.
- **PreviousCompensation** - Extracts the compensation value from the previous period data fetched in one of the previous elements. It colors the result according to the value.
- **CurrentCompensation** - Extracts the baseline value from the current period data fetched in one of the previous elements. It colors the result according to the value.
- **CompensationPayment** - Extracts the Compensation Payment value (taking holdback into account). It colors the result according to the value.
- **IsForecastApplied** - Returns the value defining whether the forecast method has been selected, therefore forecasts should be calculated.
- **ForecastMethod** - Passes the value of the Forecast Method input.
- **SalesGoalIncreasePct** - Passes the value of Sales Goal Increase % input.
- **ForecastBaselineValue** - Extracts the forecasted baseline value from the current period data fetched in one of the previous elements.
- **ForecastCompensation** - Extracts the forecasted compensation value from the current period data fetched in one of the previous elements.
- **TrueUp** - Forecast Compensation minus Current Compensation.
- **QuotaFor** - Retrieves the quota for a value stored in the configuration of a currently processed Compensation Type of Advanced Configuration of the package (default). Can be either Payment Period or Plan Duration.
- **DepositScheme** - Retrieves the deposit scheme value stored in the configuration of a currently processed Compensation Type of Advanced Configuration of the package (default). Can be either Cumulative or Non-Cumulative.
- **PaymentPeriod** - Passes the value of the Payment Period input.
- **SellerNamesTable** - Passes the value of the Seller input, stores the name of the Sellers.
- **CustomerGroup** - Passes the value of the CustomerGroup input.
- **ProductGroup** - Passes the value of the ProductGroup input.

- **GeneralFilter** - Passes the value of the GeneralFilter input.
- **PayoutCurrency** - Passes the value of the Payout Currency input.
- **CurrentPeriods** - Returns a list of periods based on the line item payment period input and CO plan startDate/endDate.
- **PreviousPeriods** - Returns a list of previous periods based on the line item payment period input and CO plan startDate/endDate.
- **PreviousCompensationDataOnPeriods** - Returns data (adjustment data & Datamart data) of the previous periods. The period is based on the payment period input, used later on in calculations.
- **CurrentCompensationDataOnPeriods** - Returns data (adjustment data & Datamart data) of the current periods. The period is based on the payment period input, used later on in calculations.
- **PreviousMonthlyPeriodsData** - Returns data (adjustment data & Datamart data) of the previous monthly periods, used to aggregate data in the header.
- **CurrentMonthlyPeriodsData** - Returns data (adjustment data & Datamart data) of the current monthly periods, used to aggregate data in the header.
- **AddingMonthlyPeriodsDataToCache** - Adds previous monthly periods data and current monthly periods data to the cache, used to pass data to the header (post phase) for aggregation header data.
- **ForecastCompensationComparisonChart** - Bar chart displaying Previous Compensation vs. Current Compensation vs. Forecasted Compensation.
- **ForecastBaselineValueTable** - Table displaying this year's sales forecast by Payment Period of the particular line item.
- **ForecastBaselineValueChart** - Bar chart displaying this year's sales forecast by Payment Period of the particular line item.
- **Holdback** - Returns the holdback value that was taken into account during calculations.
- **PreviousPeriodAdjustmentIds** - Returns the list of IDs of Adjustments that were taken into account during calculations. These Adjustments are taken from the previous period.
- **CurrentPeriodAdjustmentIds** - Returns the list of IDs of Adjustments that were taken into account during calculations. These Adjustments are taken from the current period.
- **SourceCurrencyExchangeRate** - Retrieves the currency exchange rate based on the selected Payout Currency and the Datamart Currency. The Exchange rate is based on the payout date of the record.
- **SourceCurrencyPreviousBaselineValue** - Extracts the baseline value from the previous period data fetched in one of the previous elements, but in source currency.
- **SourceCurrencyCurrentBaselineValue** - Extracts the baseline value from the current period data fetched in one of the previous elements, but in source currency.
- **SourceCurrencyPreviousCompensation** - Extracts the compensation value from the previous period data fetched in one of the previous elements, but in source currency. It colors the result according to the value.
- **SourceCurrencyCompensationPayment** - Extracts the Compensation Payment value (taking holdback into account), but in the source currency.
- **SourceCurrencyCurrentCompensation** - Extracts the baseline value from the current period data fetched in one of the previous elements, but in source currency. It colors the result according to the value.
- **SourceCurrencyHoldback** - Returns the holdback value that was taken into account during calculations, but in the source currency.
- **SourceCurrencyForecastBaselineValue** - Extracts the forecasted baseline value from the current period data fetched in one of the previous elements, but in source currency.
- **SourceCurrencyForecastCompensation** - Extracts the forecasted compensation value from the current period data fetched in one of the previous elements, but in source currency.
- **SourceCurrencyTrueUp** - Forecast Compensation minus Current Compensation in source currency.
- **SourceCurrency** - Defines the Source currency of the source data, the same as DatamartCurrency.
- **CompensationRecords** - Generates the Compensation Record shells according to the processed information. This element is only run in the Plan context.

- **RegenerateCompensationRecords** - Re-generates the Compensation Records if any of the re-generation conditions were met. This element is run only in the Generate Compensation Record context.
- **GenerateAccrualRecords** - Generates the main calculation data snapshot in the Accrual Records table. This element is run only in the Generate Accrual Record context.
- **CalculationBase** - Returns the calculationBase for the Compensation Record.

SC_Dashboard_SellerOverview_Configurator

This is the configurator logic for the Compensations Dashboard. Its main purpose is to allow selection of different Compensation Agreements based on the selected Seller.

See also the [architecture documentation](#).

Elements Description

- **SellerInput** - Presents the user with a Seller input if the user is a Sales Manager (allows to select one of the Sellers out of all who report to that person) or if the user is not a Sales Manager (a text with what Seller is being used for further processing).
- **CompensationInput** - Provides a Compensation Agreement selection option based on the Seller input defined above. The agreements are fetched in YTD period and only approved ones are considered.
- **SellerInput** - Presents the user with a Seller input if the user is a Sales Manager (allows to select one of the Sellers out of all who report to that person) or if the user is not a Sales Manager (a text with what Seller is being used for further processing).
- **ShouldAbortCalculation** - Sets a hidden input that notifies the dashboard logic to abort generation of data (performance reasons).
- **AbortInputGeneration** - Aborts further input generation in case the ShouldAbortCalculation flag is set.
- **CompensationPlanInput** - Allows the selection of plans assigned to the seller selected in Seller Input.
- **PayoutCurrencyInput** - Allows the selection of Payout Currency based on Seller and Plan selected previously.
- **YearInput** - Allows the selection of Validity Year for plans based on Seller, Plan and Payout Currency selected previously.

SC_Dashboard_SellerOverview

This is a logic for the Seller Overview dashboard. It handles display of all inputs, validation of the inputs, data fetching as well as processing of the obtained data. The dashboard displays three portlets: one data table and two charts.

See also the [architecture documentation](#).

Elements Description

- **DashboardUtils** - Contains all methods related to creation of portlets and data processing related to them. The utils are for all three currently visible portlets.
- **Configuration** - Fetches the Advanced Configuration options for the package. The most important part for the dashboard is the Datamart configuration.
- **FeatureConfig** - Fetches the FeatureConfig part of the solution that defines which optional features are turned on and off.
- **Configurator** - Displays the inline configurator defined by SC_Dashboard_SellerOverview_Configurator logic.
- **AbortIfSyntaxCheck**
- **InputValidation** - Performs the validation of any entered inputs, currently the checks are if the Seller and Payout Year is selected.
- **SourceCurrencyCode** - Retrieves the currency code of the Source Datamart defined in Configuration.

- **CompensationPlanData** - Retrieves a complex data structure containing all the data necessary for the displayed portlets. The data is in raw format, each portlet processes the data as required.
- **YTDCompensationPlanTable** - Prepares the data to be displayed in the summary Compensation Plan Table and builds the table itself. Adds necessary Context Linking. The table changes depending on the Holdback feature being enabled.
- **TotalCompensationPortlet** - Prepares the data to be displayed in the Compensation Pie chart and builds the chart itself. The chart changes depending on the Holdback feature being enabled.
- **CompensationByMonthPortlet** - Prepares the data to be displayed in the Compensation By Month column chart and builds the chart itself. The chart changes depending on the Holdback feature being enabled.
- **MixpanelTracking** - Gathers information about the usage of the dashboard for Mixpanel.

SC_Dashboard_SellerOverview_CompensationReferenceTable_PlanDetails_ContextLink

This is a logic for the Context Link part in Compensation Reference Table, for the button that leads to Plan Details.

See also the [architecture documentation](#).

Elements Description

- **ContextLink** - The element reads the passed parameters and converts them to the target page data.

SC_Dashboard_SellerOverview_CompensationReferenceTable_TransactionDetails_ContextLink

This is a logic for the Context Link part in Compensation Reference Table, for the button that leads to Transaction Details.

See also the [architecture documentation](#).

Elements Description

ContextLink - The element reads the passed parameters and converts them to the target page data.

SC_Dashboard_TransactionsAndStatistics_Configurator

This is the configurator logic for the Transactions and Statistics dashboard. Its main purpose is to allow filtering the transaction list present in the Dashboard, currently the supported inputs are: SellerGroup, Plan Currency, Compensation Plan Type, Compensation Plan Status, Compensation Plan, Compensation Record, Start Date, End Date.

See also the [architecture documentation](#).

Elements Description

- **ConfiguratorCommonUtils** - Provides util methods to read values from configurator entries or create them.
- **WarningElement** - Dummy element returning an empty ConfiguratorEntry, used to set warning messages in case of no provided inputs.
- **IsInitialRenderElement** - Creates a Hidden entry that defines whether this or any other configurator refresh is initial.
- **IsInitialRenderValue** - Reads the value that states whether this pass is an initial render, used to set up default values.
- **SellerGroupInput** - Generates the Seller Group picker input. The input is read only for Sales Agents.
- **ShouldAbortCalculation** - Creates a Hidden entry that is used to pass information to dashboard logic whether the calculation of the dashboard should be aborted.

- **AbortInputGeneration** - Aborts the generation of inputs if the condition to display them is not fulfilled.
- **PlanCurrencyInput** - Generates the Plan Payout Currency option input. This input is influenced by all inputs above.
- **CompensationPlanTypeInput** - Generates the Plan Type options input. This input is influenced by all inputs above.
- **CompensationPlanStatusInput** - Generates the Plan Status options input. This input is influenced by all inputs above.
- **CompensationPlanInput** - Generates the Plan Unique Name options input. This input is influenced by all inputs above.
- **CompensationRecordInput** - Generates the Compensation Record Unique Name options input. This input is influenced by all inputs above.
- **StartDateInput** - Generates the Start Date date entry input.
- **EndDateInput** - Generates the End Date date entry input.
- **HiddenWarningInputCheck** - Checks if any of the inputs is populated and if not, sets the warning message on the dummy WarningElement result.
- **IsInitialRenderSetElement** - Sets the value of IsInitialRenderElement to true after all the initial configurator render passes finished.

SC_Dashboard_TransactionAndStatistics

This is a logic for the Transactions and Statistics dashboard. It handles display of all inputs, validation of the inputs, data fetching as well as processing of the obtained data. The dashboard displays three portlets: one summary and two tables.

See also the [architecture documentation](#).

Elements Description

- **Configuration** - Fetches the Advanced Configuration options for the package. The most important part for the dashboard is the Datamart configuration.
- **ConfiguratorUtils** - Contains methods that are used to fetch data from the dashboard input configurator.
- **CompensationAgreementDataUtils** - Contains methods that are used for processing of data related to Compensation Plan, such as finding plans with appropriate filters.
- **TransactionUtils** - Contains methods related to transaction data, queries and processing. This is the element where the query generation happens.
- **DataTableUtils** - Contains methods related to ResultMatrix and the cell management, such as creating link cells.
- **Configurator** - Displays the inline configurator defined by SC_Dashboard_TransactionsAndStatistics_Configurator logic.
- **SyntaxCheckAbort**
- **CalculationAbort** - Safety element that prevents calculation of any other elements when certain conditions are fulfilled. The purpose is to prevent loading a dashboard with a lot of data.
- **SourceCurrency** - Returns the Datamart currency for the Datamart defined in Advanced Configuration.
- **CompensationPlanStatistics** - Displays an HTML portlet that provides the summary of the data displayed on the dashboard, such as Number of Transactions or Sellers.
- **TransactionsTable** - Displays a ResultMatrix table portlet that contains the list of transactions that are matching the provided filters.
- **CompensationRecordsTable** - Displays a ResultMatrix table portlet that contains the list of Compensation Records that are matching the provided filters.
- **MixpanelTracking** - Gathers information about the usage of the dashboard for Mixpanel.

SC_Dashboard_AdminYTDSummary_Configurator

This is the configurator logic for the Admin YTD Summary dashboard. Its main purpose is to allow filtering in the Dashboard, currently the supported inputs are: SellerGroup, Reporting Currency, Plan Currency.

See also the [architecture documentation](#).

Elements Description

- **SellerGroupInput** - Presents the user with a Seller input if the user is a Sales Manager (allows to select one of the Sellers out of all who report to that person) or if the user is not a Sales Manager (a text with what Seller is being used for further processing).
- **ShouldAbortCalculation** - Sets a hidden input that notifies the dashboard logic to abort generation of data (performance reasons).
- **AbortInputGeneration** - Aborts further input generation in case the ShouldAbortCalculation flag is set.
- **ReportingCurrencyInput** - Displays the Reporting Currency radio button, based on whether the PlanCurrencyInput is shown or not.
- **PlanCurrencyInput** - Displays the Plan Currency input if the Reporting Currency is set to Payout Currency. The available values are based on the previous selections.

SC_Dashboard_AdminYTDSummary

This is a logic for the Admin YTD Summary dashboard. It handles display of all inputs, validation of the inputs, data fetching as well as processing of the obtained data. The dashboard displays two portlets: one summary and one table.

See also the [architecture documentation](#).

Elements Description

- **DashboardUtils** - Contains various methods that are used throughout the dashboard logic, for example safe extraction of input values.
- **FeatureConfig** - Fetches the FeatureConfig part of the solution that defines which optional features are turned on and off.
- **Configuration** - Fetches the Advanced Configuration options for the package. The most important part for the dashboard is the Datamart configuration.
- **Configurator** - Displays the inline configurator defined by SC_Dashboard_AdminYTDSummary_Configurator logic.
- **SyntaxCheckAbort**
- **InputValidation** - Performs the validation of any entered inputs, currently it checks if the user has any SC User Groups assigned and if the calculation has not been aborted.
- **CompensationRecordsData** - Retrieves a complex data structure containing all the data necessary for the displayed portlets. The data is in raw format, each portlet processes the data as required.
- **CompensationsYTDSummary** - Processes the data and displays it in format of an HTML Summary portlet.
- **PayoutsOverviewTable** - Processes the data and displays it in format of a Result Matrix table.
- **MixpanelTracking** - Gathers information about the usage of the dashboard for Mixpanel.

SC_Dashboard_AdminYTDSummary_PayoutsOverviewTable_PayoutDateDetails_ContextLink

This is a logic for the Context Link part in Payouts Overview Table, for the button that leads to Payout Date Details on SC_PayoutsAndPlans Dashboard.

See also the [architecture documentation](#).

Elements Description

- **ContextLink** - The element reads the passed parameters and converts them to the target page data.

SC_Dashboard_AdminYTDSummary_PayoutsOverviewTable_PayoutSellerDetails_ContextLink

This is a logic for the Context Link part in Payouts Overview Table, for the button that leads to Seller and Payout Date Details on SC_PayoutsAndPlans Dashboard.

See also the [architecture documentation](#).

Elements Description

- **ContextLink** - The element reads the passed parameters and converts them to the target page data.

SC_Dashboard_AdminYTDSummary_PayoutsOverviewTable_SellerYearlyOverview_ContextLink

This is a logic for the Context Link part in Payouts Overview Table, for the button that leads to Seller YTD Details on SC_SellerOverview Dashboard.

See also the [architecture documentation](#).

Elements Description

- **ContextLink** - The element reads the passed parameters and converts them to the target page data.

SC_Dashboard_PayoutsAndPlans_Configurator

This is the configurator logic for the Payouts and Plans dashboard. Its main purpose is to allow filtering in the Dashboard, currently the supported inputs are: SellerGroup, Reporting Currency, Plan Currency, Payout Year and PayoutDate.

See also the [architecture documentation](#).

Elements Description

- **ConfiguratorUtils** - Contains methods that are used throughout the Configurator logs, for example to fetch common data or read inputs.
- **SellerGroupInput** - Presents the user with a Seller input if the user is a Sales Manager (allows to select one of the Sellers out of all who report to that person) or if the user is not a Sales Manager (a text with what Seller is being used for further processing).
- **ShouldAbortCalculation** - Sets a hidden input that notifies the dashboard logic to abort generation of data (performance reasons).
- **AbortInputGeneration** - Aborts further input generation in case the ShouldAbortCalculation flag is set.
- **SellerIdsInput** - Hidden input that translates the SellerGroup into a list of Seller IDs for further processing.
- **ReportingCurrencyInput** - Displays the Reporting Currency radio button, based on whether the PlanCurrencyInput is shown or not.
- **PlanCurrencyInput** - Displays the Plan Currency input if the Reporting Currency is set to Payout Currency. The available values are based on the previous selections.
- **PayoutYear** - Displays the Payout Year input that allows to narrow down the displayed values into a given year. Values available based on previous selections.
- **PayoutDate** - Displays the Payout Date input that allows to further narrow down the displayed values into a given date. Values available based on previous selections.

SC_Dashboard_PayoutsAndPlans

This is a logic for the Payout and Plans dashboard. It handles display of all inputs, validation of the inputs, data fetching as well as processing of the obtained data. The dashboard displays two portlets: one summary and two tables.

See also the [architecture documentation](#).

Elements Description

- **DashboardUtils** - Contains methods that are used to fetch data for the dashboard as well as process and prepare it for the use in portlets.
- **Configuration** - Fetches the Advanced Configuration options for the package. The most important part for the dashboard is the Datamart configuration.
- **Configurator** - Displays the inline configurator defined by SC_Dashboard_TransactionsAndStatistics_Configurator logic.
- **SyntaxCheckAbort**
- **InputValidation** - Performs the validation of any entered inputs, currently the checks are if the user has any of the SC groups assigned and if the Payout Date is selected.
- **SelectedSellers** - Returns the value of the selected seller ids from the Configurator.
- **SelectedPayoutYear** - Returns the value of the selected payout year from the Configurator.
- **SelectedPayoutDate** - Returns the value of the selected payout date from the Configurator.
- **PayoutCurrency** - Returns the currency in which the currently processed data is being payed out.
- **ActiveCompensationData** - Returns the found Active Compensation Plans data based on the provided filters. To be used in ActiveCompensationsTable portlet.
- **ActiveCompensationsTable** - Displays the Active Compensations Table Result Matrix portlet.
- **CompensationDetailsDashboard** - Opens the SC_EmbeddedCompensationDetailDashboard which provides additional details related to the currently selected plan. This dashboard opens on click on a row in ActiveCompensationsTable portlet.
- **PayoutData** - Returns the filtered data used by the dashboard, considers Seller Ids, Payout Date and Payout Currency as main data filters. Uses Datamart configuration.
- **PayoutTable** - Prepares the Result Matrix table that displays the Payouts.
- **TransactionDashboard** - Currently not used.
- **PayoutSummary** - Prepares the summary of the Payouts. Contains information about how many sellers are being processed, the number of compensation plans and number of payouts.
- **MixpanelTracking** - Gathers information about the usage of the dashboard for Mixpanel.

SC_Dashboard_PayoutsAndPlans_PayoutsTable_Details_ContextLink

This is a logic for the Context Link part in Payouts Table, for the button that leads to Payout Details on SC_TransactionsAndStatistics dashboard.

See also the [architecture documentation](#).

Elements Description

- **ContextLink** - The element reads the passed parameters and converts them to the target page data.

SC_EmbeddedCompensationDetailDashboard

This is a logic for the embedded dashboard that is used in Payee Plans Dashboard. The dashboard displays additional information about any selected plan.

See also the [architecture documentation](#).

Elements Description

- **ConfigManager** - Returns the instance of ConfigManager, which is a structure containing various configuration related information.
- **DashboardUtils** - Contains methods that are used to fetch and process all the data required by the dashboard.
- **AbortSyntaxCheck**
- **CompensationDetailTable** - Displays the details table based on the plan that has been selected in the Payee Plans Dashboard.

SC_Compensation_RecordCalculationFeeder

This is a feeder logic used for fetching and providing items for scheduled compensation calculation executions.

See also the [architecture documentation](#).

Elements Description

- **SyntaxCheckAbort**
- **EmittingCompensationRecord** - Main element that emits the Compensation Records based on the provided filters.

SC_AdjustmentHeader

This is a header logic for the Adjustment Custom Form object. It is used to read the properties of the object and pass them to the header configurator.

See also the [architecture documentation](#).

Elements Description

- **Configurator** - Initializes the header configurator in the customFormProcessor and passes appropriate parameters to it.

SC_AdjustmentHeaderConfigurator

This is the main Adjustment object configurator that is used to generate adjustments based on user provided inputs.

See also the [architecture documentation](#).

Elements Description

- **DatamartName** - Returns the Datamart name stored in Advanced Configuration.
- **SellerData** - Returns all Sellers (with labels) stored in the system.
- **AdjustmentFormId** - Retrieves the adjustment ID value passed from the header logic.
- **CreatedByInput** - Returns a read-only text input that displays the Created By value.
- **CreatedOnInput** - Returns a read-only text input that displays the Created On value.
- **TitleInput** - Returns a read-only text input that displays the Adjustment title.
- **StatusInput** - Returns a read-only text input that displays the current Adjustment status.
- **DescriptionInput** - Returns a text input that allows the user to insert a description for the Adjustment.
- **ProductGroupInput** - Returns a Product Group selection input used to filter out transactions affected by the Adjustment.
- **CustomerGroupInput** - Returns a Customer Group selection input used to filter out transactions affected by the Adjustment.

- **SellerInput** - Returns a Seller selection input used to filter out transactions affected by the Adjustment.
- **PaymentDateInput** - Returns a Payment Date selection input used to filter out transactions affected by the Adjustment.
- **GeneralFilterInput** - Returns a General Filter input used to filter out transactions affected by the Adjustment.
- **TransactionsDataMatrix** - Displays the transaction matrix that shows transactions that will be affected by the adjustment. The display limit is set to show top 10 transactions, the rest is hidden due to performance reasons.
- **AdjustmentTypeInput** - Returns an option input that allows the selection of the Adjustment type that will define how the adjustment will be generated and handled.
- **NewSellerInput** - In case of "Change Data" Adjustment type allows to insert a new Seller value for the selected transactions.
- **NewProductInput** - In case of "Change Data" Adjustment type allows to insert a new Product value for the selected transactions.
- **NewCustomerInput** - In case of "Change Data" Adjustment type allows to insert a new Customer value for the selected transactions.
- **NewDateInput** - In case of "Change Data" Adjustment type allows to insert a new Pricing Date value for the selected transactions.
- **BaselineInputsInfo** - In case of "Change Data" Adjustment type displays some additional information for the user about baseline fields.
- **NewInvoicePriceInput** - In case of "Change Data" Adjustment type allows to insert a new Invoice Price value for the selected transactions.
- **NewQuantityInput** - In case of "Change Data" Adjustment type allows to insert a new Quantity value for the selected transactions.
- **NewGrossMarginInput** - In case of "Change Data" Adjustment type allows to insert a new Gross Margin value for the selected transactions.
- **NewSellersCreditSplit** - In case of "Split Credit" Adjustment type displays an Input Matrix that allows the user to define the Seller credit split. If the split percentage does not equal to 100%, an error message is shown.
- **DataTypeInput** - Allows selection of either Manual or Transaction data type adjustment.
- **SelectedDataType** - Reads the value selected in the DataTypeInput, used for conditionally hiding elements that are not valid for the given type.
- **ManualDataInfoInput** - Read-only input that displays additional information to the user about the behaviour of the Manual type.
- **ManualDataMatrixInput** - Displays the InputMatrix input that allows the user to enter the manual data to be used for Adjustments.

SC_AdjustmentWizardExecutor

This is an executor logic for the Adjustment Configurator Wizard, used to create adjustment objects.

See also the [architecture documentation](#).

Elements Description

- **CreateAdjustmentCFO** -- Generates the adjustment object based on the provided inputs and stores the appropriate values in attributes.
- **SummaryScreen** -- Displays a summary message informing the user what happened after submission of the Adjustment.

SC_AdjustmentWizardInputs

This is an input logic for the Adjustment Configurator Wizard, used to display inputs for the user.

See also the [architecture documentation](#).

Elements Description

- **SellerData** – Returns all Sellers (with labels) stored in the system.
- **TitleInput** – Generates the text input that allows the user to insert the title of the adjustment.
- **DescriptionInput** - Generates the text input that allows the user to insert the description of the adjustment.
- **ProductGroupInput** – Generates the Product Group input that allows the user to provide transactions for the Adjustment.
- **CustomerGroupInput** – Generates the Customer Group input that allows the user to provide transactions for the Adjustment.
- **SellerInput** – Generates the Seller options selection input that allows the user to provide transactions for the Adjustment.
- **PaymentDateInput** – Generates the Date input that allows the user to provide transactions for the Adjustment.
- **GeneralFilterInput** – Generates the General Filter input that allows the user to provide transactions for the Adjustment.
- **DataTypelInput** - Allows selection of either Manual or Transaction data type adjustment.
- **SelectedDataType** - Reads the value selected in the DataTypelInput, used for conditionally hiding elements that are not valid for the given type.

SC_QuotingPluginLib

This is a library that is used for integration with quoting solutions. It allows you to inject the Sales Compensation into the calculation results. The library also supports displaying the results in a pie chart or line score chart.

See also the [architecture documentation](#).

Elements Description

- **ConstConfig** – Contains definitions of global constant variables that are used throughout the logic. Any configuration-like constants should be stored here to avoid hard-coding and to allow for better maintainability of the solution
- **ConfigManager** - Strictly defined structure that allows management of solution configuration. The benefit is that the whole configuration is contained within one “class”, it cannot be modified and is cached.
- **SalesCompensationEstimation** - Stores methods used to calculate the estimated compensation based on items stored in a quote.
- **ChartUtils** - Contains methods used to generate the charts supported by the solution.

SC_NoConditionTypeFilter

This is a filtering logic for the Default Header type present on the partition. It is created so the Default type does not display any SC related Condition Types during item selection.

See also the [architecture documentation](#).

Elements Description

- **SalesCompensationFilter** - Returns a filter that filters out all Condition Types that use SC_Compensation formula.

SC_SPIFFConditionTypeFilter

This is a filtering logic for the SPIFF Header type. It is created so the SPIFF type does not display any non-SPIFF related Condition Types during item selection.

See also the [architecture documentation](#).

Elements Description

- **SPIFFTypesFilter** - Returns a filter that filters out all Condition Types that are applicable for SPIFFs.

SC_DefaultConditionTypeFilter

This is a filtering logic that should be set as default filtering when Sales Compensation Accelerator is in active use on the partition. It is created so any Sales Compensation Condition types are available during line item creation based on the Header type (Individual/Group).

See also the [architecture documentation](#).

Elements Description

- **SalesCompensationFilter** - Returns a filter that filters out all Condition Types that are applicable for either Individual or Group Header Types.

Logics Overview (Sales Compensation)

Logic Name	Default Label	Description
SC_Compensation	Sales Compensation	This logic processes all compensation types. If you need to add more input/output elements for the compensation, add them in this logic.
SC_CompensationHeader	Sales Compensation Header	Header logic for a personal compensations.
SC_CompensationRecordCalculationFeeder	Compensation Record Calculation Feeder	Finds and calculates agreement records.
SC_Dashboard_SellerOverview		Input generation, portlet generation, data fetching, etc.
SC_Dashboard_SellerOverview_Configurator	Seller Overview	Allows selection of different Compensation Agreements based on the selected Seller.
SC_OverviewReport	Overview Report	Logic supporting report of the same name.
SC_SalesCompensationAgreementReport	Sales Compensation Agreement Report	Logic supporting report of the same name.
SC_TransactionsReport	Transactions Report	Logic supporting report of the same name.

SC_YTDReport	Year To Date Report	Logic supporting report of the same name.
--------------	---------------------	---

Core Logics

Logic Name	Description
SC_CompensationHeader	Header logic for the Sales Compensation header. Contains user inputs (Seller, Currency, General Filter) and outputs (Current/Previous Compensation).
SC_Forecast_Configurator	Header configurator for Forecast Input. Displays two inputs: Forecast Method and Sales Goal Increase Percent.
SC_CompensationHeaderSellerConfigurator	Header configurator for Seller Input. Displays additional information from Seller Master for singular Seller.
SC_Compensation	Main calculation logic for the package. Handles all possible calculation contexts.
SC_QuotingPluginLib	Library used as a plugin for Quoting solutions, allows you to display compensation values during quote calculations.

Backend Logics

Logic Name	Description
SC_ProcessingCommonsLibrary	Contains common functions used in the Sales Compensation package. The functions are used for various data processing.
SC_DashboardCommonsLibrary	Contains common functions used in the Sales Compensation package. The functions are used for dashboard creation and dashboard data processing.
SC_CompensationDashboardsLib	Contains functions related to dashboards, specific for the Sales Compensation package.
SC_CompensationProcessingLib	Contains functions related to processing, specific for the Sales Compensation package.
SC_CompensationTypesLib	Each element in this library describes one supported Compensation Type. Each type has a calculation method and specific parameters.
SC_CompensationRecordCalculationFeeder	Feeder logic for Calculation Plan, selects which records should be included in the calculation run.

Adjustments Logics

Logic Name	Description
SC_AdjustmentHeader	

	Header logic for Adjustment Custom Form. Contains definition of the configurator and it passes parameters.
SC_AdjustmentHeader_Configurator	Main configurator logic for the Adjustment Custom Form object. Displayed in the Query Details tab. Handles the whole setup of an Adjustment object.
SC_AdjustmentWizard_Executor	Executor logic for Adjustment Configurator Wizard, creates the Adjustment Custom Form Object.
SC_AdjustmentWizard_Inputs	Displays inputs of the Adjustment Custom Form.

Output Logics

Logic Name	Description
SC_Dashboard_SellerOverview	Both these logics are responsible for the Seller Overview dashboard. They handle display of inputs, data fetching and processing.
SC_Dashboard_Compensations_Configurator	
SC_Dashboard_AdminYTDSummary	Admin YTD Summary dashboard calculation logic. Handles display of inputs, data fetching and processing.
SC_Dashboard_PayoutsAndPlans	Payouts and Plans dashboard calculation logic. Handles display of inputs, data fetching and processing.
SC_Dashboard_Transactions_Configurator	Both these logics are responsible for the Transactions and Statistics dashboard. They handle display of inputs, data fetching and processing.
SC_Dashboard_TransactionsAndStatistics	
SC_Dashboard_Date_Configurator	Date configurator used in the Admin YTD Summary and Payouts and Plans dashboards. Allows selection of various Payout Date related fields.
SC_Embedded_Dashboard_Compensation_Detail	Embedded dashboard logic used in Payouts and Plans dashboard. Handles display of detail information about a selected plan.
SC_OverviewReport	Logic responsible for generation of the Overview report. Element names need to match the publishing template present in the system.
SC_SalesCompensationPlanReport	Logic responsible for generation of the Sales Compensation Agreement report. Element names need to match the publishing template present in the system.
SC_TransactionsReport	Logic responsible for generation of the Transaction List report. Element names need to match the publishing template present in the system.

SC_YTDReport

Logic responsible for generation of the YTD report. Element names need to match the publishing template present in the system.

Filter Logics

Logic Name	Description
SC_DefaultConditionFilter	Condition Type Filter Logic used to limit the visibility of Condition Types during plan selection. It returns all Condition Types that are related to the Sales Compensation package.
SC_NoConditionTypeFilter	Condition Type Filter Logic used to limit the visibility of Condition Types during plan selection. It returns all Condition Types that are not related to the Sales Compensation package.
SC_SellerInputFilter	Filter logic used in various Seller Pickers to limit the visibility of the selection based on the available data and user permissions.

Libraries (Sales Compensation)

Library Name	Default Label	Description
SC_Compensation ProcessingLib	Sales Compensation Processing Library	Provides common functions used in the compensation package such as utilities for dates manipulation, inputs, record fetching, etc.
SC_Compensation TypesLib	Condition Type Library	Defines all compensation types. When you have a new condition type, add a new element to this library to define functions for this condition type.
SC_Compensation DashboardsLib	Sales Compensation Dashboard Library	Provides common functions used in dashboards of the compensation package.
SharedLib		Provides common functions used in the compensation package. Provided by Shared Groovy Library dependency.
ApprovalWorkflow		Provides function for approval workflow logics. Provided by Approval Workflow Library dependency.
FormulaEvaluator		Used in Approval Workflow. Provided by Formula Evaluator Library dependency.
HighchartLibrary		Provides "Highcharts" functions used in dashboards. Provided by Dashboards Accelerator Library dependency.

Company Parameters (Sales Compensation)

The following supporting configuration parameters are deployed with the package and are used for configuration.

- [SC_AdjustmentWizardTransactionFieldMapping](#)
- [SC_ConditionTypes](#)
- [SC_CustomerFieldMapping](#)
- [SC_FeatureConfig](#)
- [SC_HeaderSellerDetailsMapping](#)
- [SC_ProductFieldMapping](#)
- [SC_QuotingPluginConfiguration](#)
- [Dependency Parameters](#)

SC_AdjustmentWizardTransactionFieldMapping

Column Label	Column Name	Description
Adjustment Field Name	key1	
Datamart Field Name	key2	

SC_ConditionTypes

This parameter saves the Condition Type code and Condition Type name. If the package supports a new Condition Type, you need to add a new row in this table.

The value of the "Condition Type Name" column will be shown in the "Condition Type Name" (attribute4) column from the Condition Types table from the Rebates module.

Column Label	Column Name	Description
Condition Type Code	name	Must match the element name in Groovy Library "SC_CompensationTypesLib" (and "RebateTypesLib" from Rebate Manager if deployed together).
Condition Type Name	attribute1	Will be shown in "Condition Type Name".

The screenshot shows two application windows side-by-side. The left window, titled 'Condition Types', has a search bar and a list of condition types with columns for Name, Label, and Condition Type Name. The right window, titled 'Company Parameter Values: RM & SC Condition Types', also has a search bar and a list of condition types with columns for Condition Type Code and Condition Type Name. A red box highlights the 'Condition Type Name' column in both windows, and a red arrow points from the right window to the left one, indicating that the value in the right window's 'Condition Type Name' column is used to populate the 'Condition Type Name' column in the left window's list.

SC_CustomerFieldMapping

This parameter maps the Customers master table attributes (source fields) with the Datamart field names. You should check it and update based on the master table structure in your partition.

Column Label	Column Name	Description
Customer Master Field Name	name	Attribute name (source field) in Customers
Datamart Field Name	attribute1	Map with a field name in Datamart

SC_FeatureConfig

Column Label	Column Name	Description
Feature Name	name	Possible values: HOLDBACK, FORECAST, HEADER GENERAL FILTER, LINE ITEM GENERAL FILTER
Status	attribute1	Possible values: ON, OFF

SC_HeaderSellerDetailsMapping

Column Label	Column Name	Description
Field Label	key1	
Seller Field	key2	

SC_ProductFieldMapping

This parameter maps the Products master table attributes (source fields) with the Datamart field names. You should check it and update based on the master table structure in your partition.

Column Label	Column Name	Description
Product Master Field Name	name	Attribute name (source field) in Products
Datamart Field Name	attribute1	Map with a Field Name in Datamart

SC_QuotingPluginConfiguration

Column Label	Column Name	Description
Value	attribute1	
Category	key1	
Sub Category	key2	

Key Name	key3
----------	------

Dependency Parameters

List of parameters that come from dependencies and have an effect on the package behavior.

- CurrencySymbols
- ApprovalCondition
- ApprovalWorkflowSetup
- Approvers
- WFExpressionVariableConfiguration
- WFLookupFilterConfiguration

Advanced Configuration Options (Sales Compensation)

Package configuration is done from this central point in the system. Option values are configured mostly during a package deployment process and can be adjusted later. Some options can be overridden on a Condition Type or Line Item level where the lower level has a priority.

The following table shows all configuration parameters stored in `SC_AcceleratorConfiguration`.

Name	Description
<code>sourceType</code>	
<code>sourceName</code>	Name of Datamart used to query and allocate the compensation value
<code>baselineFieldName</code>	The field name of a Value Base field is used to calculate the base value. In most cases, it will be a revenue or margin field.
<code>sellerIDFieldName</code>	The field name contains an identification of a sales agent.
<code>customerIDFieldName</code>	The field name contains a customer ID in the Datamart.
<code>productIDFieldName</code>	The field name contains a product ID in the Datamart.
<code>productNameFieldName</code>	The field name contains a product name. It is used for the <i>Transaction List</i> report only.
<code>pricingDateFieldName</code>	The field name contains a pricing date in the Datamart.
<code>currencyFieldName</code>	The field contains the main Datamart currency.
<code>payoutDays</code>	A number of days after the end date of the payment period when the payout happens. The parameter defines the default "Payout Days" for Condition Types that do not have it filled in directly. The value can be overridden in Condition Types.
<code>quotaFor</code>	The quota (target) defined on the line item is evaluated for each payment period (see <i>paymentPeriod</i>) or for the whole plan duration. The parameter defines the default "Quota For" for Condition Types that do not have it filled in directly.

	<p>Select one of these options:</p> <ul style="list-style-type: none"> • Payment Period - Gets a base value for every period, then compares and calculates the compensation. • Plan Duration - Gets a base value for the whole plan duration, then compares and calculates the compensation. <p>The value can be overridden in Condition Types.</p>
paymentPeriod	<p>A parameter setting a frequency of payouts. The parameter defines the default "Payment Period" for Condition Types that do not have it filled in directly. The user can specify it also on a Line Item level.</p> <p>Values: "Monthly", "Quarterly", "Semi-Annually", "Annually"</p> <p>The value can be overridden in Condition Types.</p>
depositScheme	<p>A parameter providing guidance on compensation accumulation across payment periods.</p> <p>If "quotaFor" is "Plan Duration", you need to define how to calculate the compensation value:</p> <ul style="list-style-type: none"> • Non-Cumulative - The compensation value is calculated for the current period based on a cumulation of the base value, then divided by the number of periods (12 months or 4 quarters, 2 for semi-annually, 1 year). • Cumulative - The same as with calculation of Non-Cumulative above, but the compensation value of the previous period is excluded. <p>The value can be overridden in Condition Types.</p>
marginFieldName	
invoicePriceFieldName	
quantityFieldName	

System Objects Attributes (Sales Compensation)

Sellers

Label	Attribute	Description
Seller ID	sellerId	Used to identify the sellers. These fields are most often used in the code.
Seller Name	name	
Reports To	reportsTo	Used in dashboards to identify a manager of a given seller. If empty, the person is the manager.
First Name		

	attribute1	Used in dashboards and selection options for more user friendly identification of sellers.
Surname	attribute2	
Pricefx User Account ID	attribute3	Used in dashboards to map a seller to a currently logged in user to restrict the visible content.
Termination Date	attribute5	Defines the termination date of the seller. This affects (re)generation of records related to the seller.

Compensation Plan

Label	Attribute	Description
Name	uniqueName	
Name without revision	rootUniqueName	
Label	label	
Workflow Status	workflowStatus	
Workflow Submitter	submittedByName	
Customer(s)	customerGroup	
Seller Name	sellerName	Propagated from Header Input, defines the seller assigned to the agreement
Status	compensationPlanStatus	
Compensation Type	compensationHeaderType	Defines the type of the agreement. Used for filtering out non-SC related agreements, for example in dashboards.
Payout Currency	attributeExtension__currency	

Compensation Type

Label	Attribute	Description
Name	uniqueName	
Header Logic	headerFormulaName	Defines the logic used for all header inputs and output handling.

Compensation Plan Workflow	workflowFormulaName	Defines the logics used for workflow handling on the agreement and record level.
Compensation Record Workflow	scRecordWorkflowFormulaName	
Filtering logic	scTypeFilterFormulaName	Defines the logics used for filtering available line items to be selected in agreements.

Compensation Record

Label	Attribute	Description
ID	uniqueName	
Set	compensationRecordSetLabel	
Source ID	sourceId	
Source ID without revision	sourceIdNoRev	
Rank	rank	
Status	status	
Compensation Plan Status	compensationStatus	Taken from Plan (or Line Item) inputs or properties.
Name	name	
Label	label	
Condition Type	compensationConditionType	Defines what type of compensation has been selected.
Customer/Customer grouping		Taken from Plan (or Line Item) inputs or properties.
Product/Product grouping		Taken from Plan (or Line Item) inputs or properties.
Compensation Type	compensationHeaderType	Taken from Plan (or Line Item) inputs or properties.
Seller Name	sellerName	Taken from Plan (or Line Item) inputs or properties.
Current Baseline Value	attribute1	Typically represents revenue, gross margin or volume in some cases. It depends on the relevant Condition Type.
	attribute2	Calculation result

Compensation Payment		
Pay To ID	attribute3	Calculation result
Current Baseline Quantity	attribute4	Calculation result
Current Baseline Value (Source Currency)	attribute5	Typically represents revenue, gross margin or volume in some cases. It depends on the relevant Condition Type.
Compensation Payment (Source Currency)	attribute6	
Previous Baseline Value (Source Currency)	attribute7	This number represents a value from one period ago. It typically represents volume, revenue, or margin depending on the relevant Condition Type.
Forecasted Baseline Value	attribute8	Typically represents forecasted revenue, gross margin or volume in some cases. It depends on the relevant Condition Type.
Compensation Forecast	attribute9	Calculation result
True-up	attribute10	Calculation result
Forecast Method	attribute11	Calculation result
Sales Goal Increase %	attribute12	Calculation result
Previous Compensation (Source Currency)	attribute13	
Forecasted Quantity	attribute14	Calculation result
General Filter	attribute15	
Payout Currency	attribute16	Calculation result
Period Name	attribute17	Calculation result
Previous Baseline Value	attribute18	Calculation result
Previous Compensation	attribute19	Calculation result
Source Currency	attribute20	

Current Period Adjustment IDs	attribute21	Calculation result
Previous Period Adjustment IDs	attribute22	
Forecasted Baseline Value (Source Currency)	attribute23	Typically represents forecasted revenue, gross margin or volume in some cases. It depends on the relevant Condition Type.
Compensation Forecast (Source Currency)	attribute24	Calculation result
True-up (Source Currency)	attribute25	Calculation result
Holdback	attribute26	
Holdback (Source Currency)	attribute27	
Current Compensation	attribute28	
Current Compensation (Source Currency)	attribute29	

Compensation Accrual Record

Label	Attribute	Description
Seller ID	attribute1	
Current Period Pricing Month	attribute2	
Current Period Pricing Quarter	attribute3	
Current Period Pricing Semi Year	attribute4	
Current Period Pricing Year	attribute5	
Current Period Pricing Baseline	attribute6	
Prior Period Pricing Month	attribute7	

Prior Period Pricing Quarter	attribute8	
Prior Period Pricing Semi Year	attribute9	
Prior Period Pricing Year	attribute10	
Prior Period Pricing Baseline	attribute11	This number represents a value from one period ago. It typically represents volume, revenue, or margin depending on the relevant Condition Type.
Penultimate Period Pricing Month	attribute12	
Penultimate Period Pricing Quarter	attribute13	
Penultimate Period Pricing Semi Year	attribute14	
Penultimate Period Pricing Year	attribute15	
Penultimate Period Pricing Baseline	attribute16	This number represents a value from two periods ago. It typically represents volume, revenue, or margin depending on the relevant Condition Type.
Product ID	attribute17	
Customer ID	attribute18	
Record Period	attribute19	
Agreement Source ID	attribute20	
Adjustment Invoice Price	attribute21	
Adjustment Quantity	attribute22	
Adjustment Margin	attribute23	
Current Period Pricing Baseline (Source Currency)	attribute24	Typically represents revenue, gross margin or volume in some cases. It depends on the relevant Condition Type.

Prior Period Pricing Baseline (Source Currency)	attribute25	This number represents a value from one period ago. It typically represents volume, revenue, or margin depending on the relevant Condition Type.
Penultimate Period Pricing Baseline (Source Currency)	attribute26	This number represents a value from two periods ago. It typically represents volume, revenue, or margin depending on the relevant Condition Type.
Source Currency	attribute27	
Payout Currency	attribute28	
Pricing Date	attribute29	

Condition Type

Condition Type attributes are needed for the Sales Compensation logic to calculate the compensation value and Compensation Records.

Label	Attribute	Description
Pricing logic	formula Name	Sets a logic to run. The value should be "Compensation Logic".
Customer Selection Level	attribute1	
Customer Filter Logic	attribute2	
Payment Period	attribute3	Defines the payment period for a Condition Type. If users do not set a value for this attribute, there is the "Payment Period" input in the Compensation Agreement with the same values. Possible values: "Monthly", "Quarterly", "Semi-Annually", "Annually" The default value can be set in Advanced Configuration Options. If needed, the value can be overridden here.
Condition Type Name	attribute4	Sets a Condition Type template for the current Condition Type. The value is set to "Condition Type Name" in Company Parameters.
Payout Days	attribute5	Enter the number of days after the end date of the payment period when the user wants the payout. The default value can be set in Advanced Configuration Options. If needed, the value can be overridden here.
Compensation Source Name	attribute6	

Compensation Base Field Value	attribute7	
Compensation Base Field Date	attribute8	
Quota For	attribute9	<p>Defines the target input for a period or for the whole plan duration.</p> <p>Possible values: "Payment Period", "Plan Duration"</p> <p>The default value can be set in Advanced Configuration Options. If needed, the value can be overridden here.</p>
Source Type	attribute10	<p>Possible values: "DataMart", "DataSource"</p>
Deposit Scheme	attribute11	<p>Defines the calculation type for a compensation value. If Quota For = "Payment Period", users do not need to set a value for this attribute.</p> <p>Possible values: "Cumulative", "Non-Cumulative"</p> <p>The default value can be set in Advanced Configuration Options. If needed, the value can be overridden here.</p>
Compensation Base Field Customer ID	attribute12	
Compensation Base Filed Product ID	attribute13	
Compensation Base Field Quantity	attribute14	
Product Filter Logic	attribute15	
Compensation Base Field Invoice Price	attribute16	
Compensation Base Field Margin	attribute17	
Compensation Base Field Seller ID	attribute18	
Is Compensation Cap Applied	attribute19	<p>Enabled by default.</p> <p>Possible values: "Yes", "No"</p>

Business Roles and Their Permissions (Sales Compensation)

To use the Sales Compensation Accelerator, you need to have the following Pricefx [user roles](#):

Sales Agent	Sales Manager	Sales Admin
<ul style="list-style-type: none"> • View Customers • View Products • View Product Details • View Custom Form • View Dashboards • Export Dashboards • Run Analytics Queries 	<ul style="list-style-type: none"> • View Customers • View Products • View Product Details • View Sellers • View Seller Extensions • Manage Custom Form • Manage Compensation Plans • Manage Compensation Records • View Dashboards • Export Dashboards • Run Analytics Queries 	<ul style="list-style-type: none"> • View Customers • View Products • View Product Details • Manage Sellers • Manage Seller Extensions • Manage Custom Form • Administer Compensation Plans • Administer Compensation Records • Administer Sales Compensations Module • Edit Global Preferences • Manage Workflows • View Dashboards • Export Dashboards • Run Analytics Queries

Compensation Types (Sales Compensation)

The following Compensation Types are used in Sales Compensation Header logic.

Name	Description
Individual Compensation	Compensation is tracked for one Sales Agent who receives all payout money.
Group Compensation	Compensation is tracked for all Sales Agents who are a part of a selected Team. Comparison of Teams and Seller groups (used in SPIFF): Team allows changes to a specific date (validity of the team, validity of seller's membership in the team, terminations etc.). Teams are defined with the help of Seller Extensions.
SPIFF (Sales Program Incentive Funds)	Compensation is tracked for all Sales Agents who are a part of a Seller group. This group is based on Master data.

Condition Types & Calculation Examples (Sales Compensation)

This section provides an overview and calculation examples for types that are distributed in the package.

- [Condition Types](#)
- [Calculation Examples](#)
- [Compensation Records Calculation](#)

Condition Types

List of available types with their default setup. Below you will find how to modify the setup.

Type	Name	Quota Type	Inputs (with units)	Notes
Unconditional	Zero Quota Percent	None	Compensation input: %	Simple % compensated to the sales agent, without any quota needed (no minimal amount of sales needed to be eligible for compensation). To be used mainly with SPIFF Compensation Plan.
Conditional	Single Quota Amount	Single	Quota input: \$ Compensation input: \$	Can be interpreted as: "if you reach X amount, you get Y amount". Available also for the SPIFF Compensation Plan.
	Single Quota Percent	Single	Quota input: \$ Compensation input: %	Available also for the SPIFF Compensation Plan.
Multi	Multi Quota Amount	Multi	Quota input: \$ Compensation input: \$	Available also for the SPIFF Compensation Plan.
	Multi Quota Percent	Multi	Quota input: \$ Compensation input: %	Available also for the SPIFF Compensation Plan.
Stepped	Repetitive Quota Amount	Multi	Quota input: \$ Compensation input: \$	Captures scenario e.g. get 10\$ compensation for every 1000\$ of revenue. To be used mainly with SPIFF Compensation Plan.
	Stepped Amount	Multi	Quota input: \$ Compensation input: \$	Available also for the SPIFF Compensation Plan.
	Stepped Percent	Multi	Quota input: \$ Compensation input: %	Available also for the SPIFF Compensation Plan.
Growth	Growth Absolute Amount	Multi	Quota input: \$ Compensation input: \$	Can be interpreted as: "if you grow by X% from Q1 to Q2, you get Y amount".
	Growth Absolute Percent	Multi	Quota input: \$ Compensation input: %	
	Growth Percent Amount	Multi	Quota input: % Compensation input: \$	
	Growth Percent Percent	Multi	Quota input: % Compensation input: %	

Bonus	Flat Bonus	Single	Quota input: \$ Target Incentive input: \$	Available also for the SPIFF Compensation Plan.
	Multi Target Bonus	Multi	Quota input: \$ Target Incentive input: \$ Multi Target Bonus: <ul style="list-style-type: none"> • Target Achievement: % • Bracket Bonus Rate: % 	Available also for the SPIFF Compensation Plan.
	Stepped Bonus	Multi	Quota input: \$ Target Incentive input: \$ Stepped Bonus input: <ul style="list-style-type: none"> • Target Achievement input: % • Bracket Bonus Rate input: % 	Available also for the SPIFF Compensation Plan.
Variable Pay	Variable Pay Linear Amount	Single	Quota input: \$ Variable Pay input: %	
	Variable Pay Multi Target Amount	Multi	Quota input: \$ Variable Pay input: % Stepped Bonus input: <ul style="list-style-type: none"> • Target Achievement input: % • Bracket Bonus Rate input: % 	
	Variable Pay Stepped Amount	Multi	Quota input: \$ Variable Pay input: % Stepped Bonus input: <ul style="list-style-type: none"> • Target Achievement input: % • Bracket Bonus Rate input: % 	

i The dollar \$ sign represents a currency in general, not an actual USD.

Calculation Examples

Zero Quota Percent

Inputs in the plan:

- Compensation - %

Success formula (always success): $(\text{COMPENSATION_INPUT} / 100) \times \text{BASELINE}$

Example

Compensation [input]	Baseline	Compensation [output]
1%	\$110,000	\$1,100
1%	\$90,000	\$900

Single Quota Amount

Inputs in the plan:

- Quota - \$
- Compensation - \$

Success formula: COMPENSATION_INPUT

Example

Quota [input]	Compensation [input]	Baseline	Compensation [output]
\$100,000	\$1,000	\$110,000	\$1,000
\$100,000	\$1,000	\$90,000	\$0

Example with Volume Based Quota

Let's assume that for every 10 products sold the Sales Agent gets 10\$. Quota can be defined as volume (Quantity field in the Datamart, can be pieces, kg, etc.). Baseline stands for how many pieces the seller sold.

Volume based quota is applicable to all amount/bonus types (types where compensation is not defined as % of baseline).

Quota [input]	Compensation [input]	Baseline	Baseline Value	Compensation [output]
10	\$10	Quantity	4	\$0
10	\$10	Quantity	15	\$10

Single Quota Percent

Inputs in the plan:

- Quota - \$
- Compensation - %

Success formula: $(\text{COMPENSATION_INPUT} / 100) \times \text{BASELINE}$

Example

Quota [input]	Compensation [input]	Baseline	Compensation [output]
\$100,000	1%	\$110,000	\$1,100
\$100,000	1%	\$90,000	\$0

Multi Quota Amount

Inputs in the plan:

- Quota - \$
- Compensation - \$

Success formula: $\text{COMPENSATION_INPUT}$

Example

Quota [tiered input]	Compensation [tiered input]
\$1000	\$100
\$1500	\$150

Result:

Baseline (\$)	Compensation [output]
\$100	\$0
\$1100	\$100
\$1600	\$150

Multi Quota Percent

Inputs in the plan:

- Quota - \$
- Compensation - %

Success formula: $(\text{COMPENSATION_INPUT} / 100) \times \text{BASELINE}$

Example

Quota [tiered input]	Compensation [tiered input]
\$1000	1%
\$1500	10%

Result:

--	--

Baseline (\$)	Compensation [output]
\$100	\$0
\$1100	\$11
\$1600	\$160

Repetitive Quota Amount

Inputs in the plan:

- Quota(s) - \$
- Compensation(s) - \$

Success formula: $\text{COMPENSATION_INPUT} \times (\text{BASELINE} / \text{QUOTA})$ (Euclidean division)

Example

Definition:

Quota [input] - Step	Compensation [input]
\$10,000	\$100

Result:

Baseline	Compensation [output]
\$5,000	\$0
\$15,000	\$100
\$110,000	\$1,100

Stepped Amount

Inputs in the plan:

- Quota(s) - \$
- Compensation(s) - \$

Success formula for step 1: $\text{COMPENSATION_INPUT_FOR_STEP1}$

Success formula for step 2: $\text{COMPENSATION_INPUT_FOR_STEP1} + \text{COMPENSATION_INPUT_FOR_STEP2}$

Example

Definition:

Quota [tiered input] - Step	Compensation [input]
\$10,000	\$100
\$50,000	\$500

\$100,000	\$5,000
-----------	---------

Result:

Baseline	Compensation [output]
\$5,000	\$0
\$15,000	\$100
\$110,000	$\$100 + \$500 + \$5,000 = \$5,600$

Stepped Percent

Inputs in the plan:

- Quota(s) - \$
- Compensation Value(s) - %

Success formula for step 1: $(\text{COMPENSATION_INPUT_FOR_STEP1} / 100) \times (\text{BASELINE} - \text{TARGET_STEP1})$

Success formula for step 2: $((\text{COMPENSATION_INPUT_FOR_STEP1} / 100) \times \text{TARGET_STEP1}) + ((\text{COMPENSATION_INPUT_FOR_STEP2} / 100) \times (\text{BASELINE} - \text{TARGET_STEP2}))$

Example

Definition:

Quota [tiere input] - Step	Compensation Value [input]
\$10,000	1%
\$50,000	3%
\$100,000	10%

Result:

Baseline	Compensation [output]
\$5,000	\$0
\$15,000	$(\$5,000 * 1\%) = \50
\$110,000	$(\$40,000 * 1\%) + (\$50,000 * 3\%) + (\$10,000 * 10\%) = \$400 + \$1,500 + \$1,000 = \$2,900$

Growth Absolute Amount

Comparison to some previous period - Month, Quarter, Semi-Year, Year.

Inputs in the plan:

- Quota(s) - \$
- Compensation Value(s) - \$

Success formula: COMPENSATION_INPUT_FOR_TIER

Example

Definition:

Quota [input] - Growth Tier	Compensation Value [input]
\$10,000	\$100
\$25,000	\$300
\$100,000	\$10,000

Result:

Growth Baseline (Current - Previous)	Compensation [output]
\$5,000	\$0
\$30,000	\$300
\$150,000	\$10,000

Growth Absolute Percent

Comparison to some previous period - Month, Quarter, Semi-Year, Year.

Inputs in the plan:

- Quota(s) - \$
- Compensation Value(s) - %

Success formula: $(\text{COMPENSATION_INPUT_FOR_TIER} / 100) \times \text{BASELINE}$

Example

Definition:

Quota [input] - Growth Tier	Compensation [input]
\$10,000	1%
\$25,000	2%
\$100,000	5%

Result:

Baseline	Compensation [output]
\$5,000	\$0
\$25,000	\$500
\$150,000	\$7,500

Growth Percent Percent

Comparison to some previous period - Month, Quarter, Semi-Year, Year.

Inputs in the plan:

- Quota(s) - %
- Compensation Value(s) - %

Success formula: $(\text{COMPENSATION_INPUT_FOR_TIER} / 100) \times \text{BASELINE}$

Example

Definition:

Quota [input] - Growth Tier %	Compensation Value [input]
2%	1%
5%	3%
10%	5%

Result:

Baseline	Calculation Baseline	Compensation [output]
1%	\$10,000	\$0
2%	\$100,000	\$1,000
11%	\$100,000	\$5,000

Growth Percent Amount

Comparison with some previous period - Month, Quarter, Semi-Year, Year.

Inputs in the plan:

- Quota(s) - %
- Compensation Value(s) - \$

Success formula: $\text{COMPENSATION_INPUT_FOR_TIER}$

Example

Definition:

Quota [input] - Growth Tier %	Compensation Value [input]
2%	\$1,000
5%	\$10,000
10%	\$25,000

Result:

Baseline	Compensation [output]
1%	\$0
2%	\$1,000
11%	\$25,000

Flat Bonus

Inputs in the plan:

- Quota - \$
- Target Incentive - \$

Success formula: $(\text{BASELINE} / \text{QUOTA}) \times \text{TARGET_INCENTIVE}$

Example

Quota [input]	Target Incentive [input]	Baseline	Compensation [output]
\$100,000	\$100	\$110,000	\$110
\$100,000	\$1,000	\$90,000	\$900

Multi Target Bonus

Inputs in the plan:

- Quota - \$
- Target Incentive - \$
- Target Achievement - %
- Bracket Bonus Rate - %

Success formula for baseline under tier 1: $(\text{BASELINE} / \text{QUOTA}) \times \text{TARGET_INCENTIVE} \times \text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100$

Success formula for baseline under tier 2: $((\text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100 \times \text{QUOTA}) / \text{QUOTA}) \times \text{TARGET_INCENTIVE} \times \text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100 + ((\text{BASELINE} - (\text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100 \times \text{QUOTA})) / \text{QUOTA}) \times \text{TARGET_INCENTIVE} \times \text{BRACKET_BONUS_RATE_FOR_TIER_2} / 100$

Example

Quota [input] - \$1,000

Target Incentive [input] - \$100

Tiers definition:

Target Achievement [tiered input]	Bracket Bonus Rate [tiered input]
120%	3%
140%	5%

250%	10%
------	-----

Result:

Baseline (\$)	Target Achievement (%)	Bracket Bonus Rate (%)	Compensation [output]
\$1,000	120%	3%	\$3
\$1,300	140%	5%	\$4.1
\$2,000	250%	10%	\$10.6
\$3,000	250%	10%	\$20.6

Stepped Bonus

Inputs in the plan:

- Quota - \$
- Target Incentive - \$
- Target Achievement - %
- Bracket Bonus Rate - %

Success formula for baseline under tier 1: $(\text{BASELINE} / \text{QUOTA}) \times \text{TARGET_INCENTIVE} \times \text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100$

Success formula for baseline under tier 2: $(\text{BASELINE} / \text{QUOTA}) \times \text{TARGET_INCENTIVE} \times \text{BRACKET_BONUS_RATE_FOR_TIER_2} / 100$

Example

Quota [input] - \$1,000

Target Incentive [input] - \$100

Tiers definition:

Target Achievement [tiered input]	Bracket Bonus Rate [tiered input]
120%	3%
140%	5%
250%	10%

Result:

Baseline (\$)	Target Achievement (%)	Bracket Bonus Rate (%)	Compensation [output]
\$1,000	120%	3%	\$3
\$1,300	140%	5%	\$5
\$2,000	250%	10%	\$10
\$4,000	above 250%	0%	\$0

⚠ Please note that the Compensation drops to zero when the tier is not defined (in the example it is the case of "above 250%" Target Achievement).

Variable Pay Linear Amount

Inputs in the plan:

- Quota - \$
- Variable Pay - %

Success formula: $(\text{BASELINE} / \text{QUOTA}) \times \text{VARIABLE_PAY} / 100 \times \text{SELLER_SALARY}$

Example

Quota [input]	Variable Pay [input]	Baseline	Seller Salary (Annual)	Compensation [output]
\$1,000,000	10%	\$300,000	\$60,000	\$1,800
\$100,000	2%	\$90,000	\$80,000	\$1,440

i In case of Group compensations the salary taken to calculations is the average of all included seller salaries.

Variable Pay Multi Target Amount

Inputs in the plan:

- Quota - \$
- Variable Pay - %
- Target Achievement - %
- Bracket Bonus Rate - %

Success formula for baseline under tier 1: $(\text{BASELINE} / \text{QUOTA}) \times \text{VARIABLE_PAY} / 100 \times \text{SELLER_SALARY} \times \text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100$

Success formula for baseline under tier 2: $((\text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100 \times \text{QUOTA}) / \text{QUOTA}) \times \text{VARIABLE_PAY} / 100 \times \text{SELLER_SALARY} \times \text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100 + ((\text{BASELINE} - (\text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100 \times \text{QUOTA})) / \text{QUOTA}) \times \text{VARIABLE_PAY} / 100 \times \text{SELLER_SALARY} \times \text{BRACKET_BONUS_RATE_FOR_TIER_2} / 100$

Example

Quota [input] - \$1,000

Variable Pay [input] - 10%

Tiers definition:

Target Achievement [tiered input]	Bracket Bonus Rate [tiered input]
120%	3%
140%	5%

250%	10%
------	-----

Result:

Baseline (\$)	Target Achievement (%)	Bracket Bonus Rate (%)	Seller Salary (\$)	Compensation [output]
\$1,000	120%	3%	\$10,000	\$3000
\$1,300	140%	5%	\$10,000	\$4,100
\$2,000	250%	10%	\$10,000	\$10,600
\$3,000	250%	10%	\$10,000	\$20,600

i In case of Group compensations the salary taken to calculations is the average of all included seller salaries.

Variable Pay Stepped Amount

Inputs in the plan:

- Quota - \$
- Variable Pay - %
- Target Achievement - %
- Bracket Bonus Rate - %

Success formula for baseline under tier 1: $(\text{BASELINE} / \text{QUOTA}) \times \text{VARIABLE_PAY} / 100 \times \text{SELLER_SALARY} \times \text{BRACKET_BONUS_RATE_FOR_TIER_1} / 100$

Success formula for baseline under tier 2: $(\text{BASELINE} / \text{QUOTA}) \times \text{VARIABLE_PAY} / 100 \times \text{SELLER_SALARY} \times \text{BRACKET_BONUS_RATE_FOR_TIER_2} / 100$

Example

Quota [input] - \$1,000

Target Incentive [input] - \$100

Tiers definition:

Target Achievement [tiered input]	Bracket Bonus Rate [tiered input]
120%	3%
140%	5%
250%	10%

Result:

Baseline (\$)	Target Achievement (%)	Bracket Bonus Rate (%)	Seller Salary (\$)	Compensation [output]
\$1,000	120%	3%	\$10,000	\$3,000
\$1,300	140%	5%	\$10,000	\$5,000

\$2,000	250%	10%	\$10,000	\$10,000
\$4,000	above 250%	0%	\$10,000	\$0

i In case of Group compensations the salary taken to calculations is the average of all included seller salaries.

w Please note that the Compensation drops to zero when the tier is not defined (in the example it is the case of "above 250%" Target Achievement).

Compensation Records Calculation

Payout Value Calculation

The "Annual" target needs a Deposit Scheme. The scheme decides if the calculation is split by the number of periods in the target time span.

Input information:

- Condition Type: [SC] Single Quota Amount
- Payment Period: Quarterly
- Quota For: **Plan Duration**

Target input:

- Target Input: 100,000.0
- Compensation Input: 1,000.0

As the payment period is each quarter, we have 4 Compensation Records with the following data:

Compensation Record	Period	Period Sale	Cumulative Sale
COR01	Q1	90,000	90,000
COR02	Q2	11,000	101,000
COR03	Q3	49,000	150,000
COR04	Q4	50,000	200,000

Non-Cumulative Example

The calculation is driven by following an execution pattern.

```

If ( Cumulative_Sale > Target_Input ) then
    Compensation_Value = Compensation_Input / 4
end

```

i Note: Number 4 stands for a number of Compensation Records.

Compensation Record	Period	Compensation Value
---------------------	--------	--------------------

COR01	Q1	0
COR02	Q2	250
COR03	Q3	250
COR04	Q4	250

Cumulative Example

The calculation is driven by following an execution pattern.

```

If ( Cumulative Sale > Target_Input ) then
    Compensation_Value = ( Compensation_Input / 4 ) *
Current_Period_Index - Accrual
end

```

Notes:

- Number 4 stands for a number of Compensation Records.
- Current Period Index starts with 1.
- Accrual is the cumulative compensation.

Compensation Record	Period	Current Period Index	Accrual	Compensation Value
COR01	Q1	1	0	0
COR02	Q2	2	0	500
COR03	Q3	3	500	250
COR04	Q4	4	750	250

Adjustments (Sales Compensation)

Adjustments allow you to modify the value of compensation that is going to be paid to a Sales Agent.

There are two places you can adjust:

- [Request Adjustment](#)
- [Adjustment Form](#) - only for the [SC] Administrator and [SC] Sales Manager roles

Request Adjustment

This Request Adjustment can be viewed by any of these: Sales Agent, Sales Manager, Administrator.

1. Go to **Pricefx Processes > Request Adjustment**.
2. Open the page.

3. Fill the required Title and Description.
4. You select which transactions you want to adjust by using these filters:
 - Product Group
 - Customer Group
 - Payee(s) (= Sales Agents or Sellers)
 - Pricing Date
 - General Filter
5. Click the **Apply** button.
6. The submit message, including your adjustment request reference number, is displayed later.

Adjustment Form

i Access to this Adjustment Form is restricted to users with the user group SC_Administrator (label: [SC] Administrator), SC_SalesManager (label: [SC] Sales Manager).

1. Go to **Sales Compensations > Adjustments**.
2. Create New Adjustments and fill in the Label and User Group (Edit) and User Group (View Details).

The screenshot shows the 'pricefx Sales Compensations / Adjustments' interface. A table lists various adjustments with columns for Name, Label, User Group (Edit), User Group (View De...), Created, Created By, and Last Update. A red box highlights the '+ Create New Adjustments' button in the top right. A modal window titled 'Create New Adjustments' is open in the center, containing three search fields: 'Label', 'User Group (Edit)', and 'User Group (View Details)', each with a question mark icon. At the bottom of the modal are 'Add' and 'Cancel' buttons.

3. Fill in option on the Adjustments Details page.

The screenshot shows the 'pricefx Sales Compensations / Adjustments / Test' details page. The breadcrumb trail is 'pricefx Sales Compensations / Adjustments / Test'. The page title is 'Test' with a sub-status 'Approved - No Approval Required'. A left sidebar contains navigation options: Adjustment Details (selected), Payee Profile, Transactions, Messages, Attachments, and Workflow. The main content area is titled 'Inputs' and displays the following information:

- Created By:** Tram Nguyen
- Created On:** Thu Oct 20 04:39:59 UTC 2022
- Title:** ---
- Status:** APPROVED
- Description:** ---

Below the 'Inputs' section is a 'Find Transactions' section with the instruction: 'Use these filters to specify transactions which you want to adjust.' It includes a 'Product Group' dropdown menu with the text 'Select Product(s)' and a 'Customer Group' field.

- a. Inputs:
- Created By

- Created On
 - Title - filled when you create from the Request Adjustment
 - Status
 - Description
- b. **Find Transactions** - Use these filters to specify transactions which you want to adjust:
- Product Group
 - Customer Group
 - Payee(s)
 - Pricing Date
 - General Filter
 - Filtered Transactions

pricefx Sales Compensations / Adjustments / Test

← Test ○ Draft - Submitted

Adjustment Details

- Payee Profile
- Transactions
- Messages
- Attachments
- Workflow

Find Transactions

Use these filters to specify transactions which you want to adjust.

Product Group
Select Product(s)

Customer Group
Select Customer(s)

Payee(s)

Pricing Date

General Filter
"Gross Margin" greater than or equal to 100.00

Warning: For performance reasons only last 10 transactions found are shown below. Please make sure you want to adjust ALL transactions according to your filter or provide more specific filter. Keep in mind that adjustments will be applied to each transaction separately.

Filtered Transactions

Product Id	Customer Id	Unique Id	Invoice Price	Gross Margin	Pricing
MB-0016	CD-00038	TXT__000020230675...	281.48	113.98	20: ▲
MB-0013	CD-00038	TXT__000020230675...	271.61	101.41	20:
MB-0015	CD-00048	TXT__000020230675...	272.19	104.87	20:

Note: For performance reasons only the last 10 transactions found are shown below. Make sure you want to adjust ALL transactions according to your filter or provide more specific filter.

Keep in mind that adjustments will be applied to each transaction separately.

- c. **Define Adjustment** - Select Adjustment Type and specify what you want to change. Defined adjustment will be applied to *every selected transaction individually*. There are the following Adjustment Types:

- **Change Data** - Allows you to change values, e.g., from Payee A to Payee B or from Date A to Date B.

You can change values in these fields: New Payee, New Product, New Customer, New Date Value, New Invoice Price, New Quantity, New Gross Margin.

pricefx Sales Compensations / Adjustments / Test

← Test ○ Draft - Submitted

- Adjustment Details
- Payee Profile
- Transactions
- Messages
- Attachments
- Workflow

Define Adjustment

Select adjustment type and specify what you want to change. Defined adjustment will be applied to every selected transaction individually.

Adjustment Type
Change Data

New Payee

New Product
Select Product

New Customer
Select Customer

New Date Value

Info
You can adjust any supported baseline field, but the adjustment will be applied to calculations only when given Condition Type uses selected field for compensation calculation.

New Invoice Price

New Quantity

New Gross Margin
100

- **Split Credit** - Allows you to split data among payees. Make sure the sum of split payees equals to 100%. If the sum of split payees is not equal to 100%, there is a warning message displayed.

Define Adjustment

Select adjustment type and specify what you want to change. Defined adjustment will be applied to every selected transaction individually.

Adjustment Type

Split Credit

Payee Credits Split

<input type="checkbox"/> Payee Name	Percentage
<input type="checkbox"/> Seller 002 (SC-002)	55
<input type="checkbox"/> Seller 003 (SC-003)	45

2 rows

Add



4. You can go to other tabs such as Payee Profile, Transactions to check dashboard data.

The screenshot shows the 'pricefx Sales Compensations / Adjustments / Test' interface. On the left, a sidebar contains 'Adjustment Details' with 'Payee Profile' selected. The 'Dashboard Settings' panel shows 'DATA FILTER' with 'Seller(s)' set to 'Select Sellers' and 'Reporting Currency' set to 'Source Currency'. The main area displays 'Admin YTD Summary', 'Compensations YTD Summary', and 'Present-day totals for 2023' with the following values:

Revenue	€ 204,558.01	Compensations	€ 135,586.84
Margin %	7.72 %	Total Margin	€ 15,791.83

Below this is the 'Payouts Overview' table:

Seller Id	Payout Date	Seller YTD Compensation Total	Seller / Payout Date Compensation Total	Invoice Price Baseline	Quantity Baseline	Individual Comp
SC-002	02/01/2023	130,659.70	130,659.70	1,550,880.74	0.00	130
SC-003	19/01/2023	4,577.13	0.00	0.00	0.00	
SC-003	02/01/2023	4,577.13	4,577.13	556,737.74	0.00	
SC-001	02/01/2023	350.00	350.00	0.00	0.00	
SC-006	02/01/2023	0.00	0.00	526.11	0.00	

5. Lastly, you need to click on the Workflow tab and submit it. Without this step, there is no adjustment generated in the Accrual Records.
6. You can check these generated adjustments in the Accrual Records:
 - a. Go to **Sales Compensations > Accrual Records**.
 - b. Filter Type == Adjustment && Compensation Record == your CFO ID you want to check.

The screenshot shows the 'pricefx Sales Compensations / Accrual Records' interface. The search bar contains '225' and the dropdown menu is set to 'Adjustment'. The table below shows the following data:

Compensation R...	Type	Name	Condition Type	Status
225	Adjustment	Search...	Search...	Search...
225	SourceAdjustment	225-Change Data-2D...		
225	TargetAdjustment	225-Change Data-r1...		

For more details on the Calculation Flow see [Flow Diagrams \(SC\)](#).

Holdback (Sales Compensation)

Holdback is defined in a Compensation Plan on the header level. It allows you to enter how much of the compensation should be held back from the payee and paid only in the last period of the plan. This mechanism helps you ensure that you do not overpay your Sales Agents in case of cancellations/returns etc. It is executed by the [logic SC_HoldBack_Configurator](#).

How Holdbacks Work

- Holdback is applicable per each line item in a value specified, per each payment period as defined on the line item.
- Holdback value defines the value which should be "held back" by the company - not paid out immediately (for the payment period). It is defined either in percentage of a compensation earned in the payment period or by flat value.
- Holdback value is subtracted from the final compensation value on the COR.
- If the holdback value (flat) is higher than a compensation earned, all the compensation is held back accordingly (but nothing is owed to the company).
- If the holdback is defined as a percentage, 100% is the maximum which can be defined. (It is validated during recalculation.)

- Holdbacks are still counted as compensation earned by the seller, but held back and planned to be paid out later.
- In CORs, there is the Holdback attribute.
 1. If Holdbacks are disabled globally, it shows "N/A" as a value. If, in the future, this gets enabled, there is information that this was not available/applicable at the time of calculation.
 2. **If the payment period is NOT the last payment period:**
 - a. For purpose of holdbacks we define *last payment* as either:
 - The last payment period of a given "Quota For Period" (e.g. the last payment of the year if Quota is defined for "Plan Duration" and the plan matches the calendar year etc.).
 - The last payment of an expiring plan (e.g. even though Quota is defined for a year, the plan expires in August, then August is the last payment).
 - It is the last payment because the seller is terminated/removed from a team and this is the last payment period for the given agent.
 - b. If the holdback was defined as a value:
 - Current Compensation > Holdback = Holdback
 - Current Compensation < Holdback = Current Compensation (the company cannot hold more than the value they are supposed to pay)
 - c. If the holdback was defined as % (max. 100%) = Current Compensation * Holdback
 - d. Holdback can never be greater than Current Compensation.
 - e. If Current Compensation < 0 (e.g. no sales + returns/adjustments etc.)
 - Holdback = Current Compensation
The seller owes money to the company for this month, so the "debt is lowered".
 3. **If the payment period IS the last payment period:**
 - a. Nothing is held back; the seller gets paid everything which was held back until now.
 - b. Holdback is the SUM(Compensation Payment(below)) for all previous periods (e.g. 11 months) - SUM(Current Compensation) for all previous periods (e.g. 11 months).
 - c. This calculates the final value. It is known how much the seller earned over all the previous periods vs. how much they actually got paid. This last holdback is used to even out the score.
 - d. See attached Holdbacks_examples.xlsx for examples and hopefully better understanding
- There is an attribute in CORs: *Compensation Payment*: Current Compensation - Holdback.

Approval Workflow (Sales Compensation)

The Approval Workflow Library is part of the package as a dependency. It is extended with new types for sales compensation context. For details see [Accelerate Approval Workflow Package](#).

Workflow Type	Workflow Name	Logic Name
SCA	Compensation Agreement Approval Workflow	SC_SalesCompensationAgreement
SCR	Agreement Record Approval Workflow	SC_AgreementRecord
Adjustment		

This page provides a list of variables used for building conditions for approval workflows inside the package. It provides support information for [Configuration \(Sales Compensation\)](#).

- [Compensation Agreement Approval Workflows](#)

- [Agreement Record Approval Workflows](#)

Compensation Agreement Approval Workflows

SCA is the name of Workflow Type that must be used for workflows you build on top of any Compensation Agreement. SCA must be used in ApprovalWorkflowSetup, ApprovalCondition and Approvers tables inside Company Parameters.

A list of all possible variables is available also in [condition expression values](#) from Approval Workflow Package.

Header - Label	Element Name	Example
Start Date	startDate	startDate == "2021-01-31"
End Date	endDate	endDate == "2021-01-31"
Payout Date	payoutDate	payoutDate == "2021-01-31"
Seller	seller	Seller == "SC-001"
Currency	currency	header.currency == "EUR"
Header - Compensation Detail		
Previous Compensation	previousCompensation	previousCompensation >= 10000
Current Compensation	currentCompensation	currentCompensation >= 50000
Line Item - Label		
Condition Type	compensationConditionType	compensationConditionType = "SC_SingleTrargetPercent"
Input		
Target	line.Target	"line.Target" > 100
Compensation	line.Compensation	"line.Compensation" == 2.0
Compensation %	line.Compensation %	"line.Compensation %" > 10
Payment Period	line.Payment Period	"line.Payment Period" == "Annually"
Line Item - Compensation Detail		
Previous Baseline Value	line.PreviousBaselineValue	"line.PreviousBaselineValue" > 127
Current Baseline Value	line.CurrentBaselineValue	"line.CurrentBaselineValue" > 1200
Previous Compensation	line.PreviousCompensation	

		"line.PreviousCompensation" > 1000
Current Compensation	line.CurrentCompensation	"line.CurrentCompensation" >= 1200
Line Item - Compensation Type Info		
Quota For	line.QuotaFor	"line.QuotaFor" == "Plan Duration"
Deposit Scheme	line.DepositScheme	"line.DepositScheme" == "Non-Cumulative"
Payment Period	line.PaymentPeriod	"line.PaymentPeriod" == "Monthly"
Seller Name	line.SellerName	"line.SellerName" == "John Doe"
General Filter	<not supported>	
Customer(s)	<not supported>	
Product(s)	<not supported>	

Agreement Record Approval Workflows

SCR is the name of Workflow Type that must be used for workflows you build on top of any Agreement Record. SCR must be used in ApprovalWorkflowSetup, ApprovalCondition and Approvers tables inside Company Parameters.

A list of all possible variables is available also in [condition expression values](#) from Approval Workflow Package.

When the condition is set up, use the field name and do not add any prefixes; see the table below with examples having specific meaning for this package.

Label	Element Name	Example
Current Baseline Value	attribute1	attribute3 > 1000
Pay To Id	attribute3	attribute3 == "SC-001"
Currency	attribute16	attribute16 == "EUR"
Current Compensation	attribute17	attribute17 >= 1500

Manual Upgrade Steps (Sales Compensation)

In this section, you will find all manual steps which need to be done while upgrading between versions. A description of the whole upgrade procedure is described in the [Upgrade \(Sales Compensation\)](#) section. In addition, you can also check out the latest release notes.

1.0.0 1.1.0

We do not support upgrading between these versions. 1.1.0 uses the Sales Compensations module instead of the Rebates module, so it needs to be deployed from scratch.

Sales Compensation 1.7.1

This document summarizes major improvements and fixes introduced in the Accelerate Sales Compensation Package release version.

Version	1.7.1
Release Date	Mar 30, 2023

Table of contents:

- [Fixed Issues](#)

Fixed Issues

Description	ID
In order to comply with the Bee's Knees 10.2 release, the recalculation button has been re-introduced to the Adjustment Custom Form. It is mandatory to click this button to able to proceed with the form creation.	PFPCS-6741