Pricefx

Knowledge Base

August 7, 2019

Knowledge Base Home

- How To Articles
- Questions and Answers
- Pricefx Studio
- Git Workflow for Configuration
- Configuration Packaging
- TDD4C
- PFX Changelogs
- Best Practices

How To Articles

- Getting Started with PFX
    - How to Create Datamart
    - How to Configure Price List
    - How to Configure a Calculated Field Set
    - How to Build a Live Price Grid
    - How to Configure a Quote
    - How to Configure PromotionManager
    - How to Configure RebateManager
    - How to Create a Price List Simulation
    - How to Run a Simulation in PriceAnalyzer
    - How to Create New Dashboard
    - How to Create a New Product Extension
    - How to Create a Chart
    - How to Upload CSV Data into PA
    - How to Set up Currencies
    - How to Load Units of Measure
    - How to Build a New Excel Template (for Quote)
    - How to Create a Matrix Price List
    - How to Write a Logic in Groovy
    - How to Configure Change History for PM or QC
- Advanced Usage Tips
    - How to Query a Data Source Directly
    - How to Transfer All Data at Once between Two Partitions
    - How to Automatically Archive Old Data
    - How to (Re)Create Missing Default PA Data Loads
    - How to Import PX Data to a Datamart
    - How to Work with Inputs in PA Formulas
    - How to Create Revenue Waterfall Dashboard
    - How to Add a Gauge Chart to Line Detail and Header
    - How to Add a Matrix to Quote Header Output Fields
    - How to Add a Quote Link to Dashboard
    - How to Debug a Price List
    - How to Add Entry to Matrix PP
    - How to Install .NET 3.5 (Required for Excel Client) with Windows 10
    - How to Run Excel Client on Mac
    - How to Find Set of Quotes Containing Given Products
    - How to Generate Excel Reports from Calculation Logic
    - How to Reference Values Between Two Price Parameter Columns
    - How to Populate Customer or Product Picker Created via API
    - How to Store Quote Inputs/Outputs in PriceAnalyzer
    - How to Add Inputs to Product Selector
    - How to Add Approver to Workflow Email Template
    - How to Create Parametric Approval Workflow
    - How to Create Button in Email Template
    - How to Use Secondary Key to Create Matrix Price Lists
    - How to Create Cascading Filters
    - How to Create Portlet Alternating Between Chart and HTML
    - How to Create Quote Line Items from Header Logic
    - How To Generate Approval Required E-mail On Quote Save
    - How To Disable Submit Quote Button in QuoteConfigurator
    - How to Hide, Disable or Make Required Fields/Buttons
    - How to Update Object References Regularly
    - How to Set Up a Buy-One-Get-One-Free Promotion
    - How to Automatically Organize Products into Folders
    - How to Set Data with Null Values in FlexCharts?
    - How to Get Quote's Effective and Expiry Dates using Calculation Logic
    - FlexChart Examples
    - How to Add Your Own Button to Navigation

Getting Started with PFX

How to Create Datamart

Creation of a Datamart consists of the following general steps:

1. Create a Data Source.
    - It must have a primary key.
    - Once deployed, the system automatically creates a Data Feed which loads data into the Data Source.
2. Create a Datamart.
    - Select the primary Data Source.
    - Pick columns from that Data Source and place them into the Datamart (by drag&drop).
    - If you want to join other, “secondary” Data Sources, the rule is: the name of the primary key column of those other Data Sources must already exist in your Datamart. (Also, the Datamart column which is used for such join must be set as a dimension.)
3. Populate all the Data Sources with data.
    - The standard way is to have data in Data Feed and using the Data Load event 'Flush' upload the data from Data Feed to Data Source.
    - Or if you have data in Excel, you can upload it directly into Data Source (or alternatively into Data Feed and then use the Data Load to copy from Data Feed into Data Source).
4. Run the Data Load of Datamart (this is called “refresh”). It will load all rows from the primary Data Source and then also the columns from other Data Sources.

⚠ Datamart is a physical table but it **does not** store a copy of all attribute values. Only keys and calculated columns are stored, the rest of the data is joined dynamically when a query is run.

Datamart (DM) is a data table, with data coming generally from one or more Data Sources (it is possible to make a Datamart without a Data Source). This data set can then be enriched:

- with data from other joined data sources (through related matching columns). Datamart is then a combination of joined data sources
- or with data from fields calculated on the fly
- or with columns, whose values are calculated by a logic (via a special Calculation Data Load)

How to Configure Price List

First, make sure you have:

- Products
- (optional) Price Parameters
- (optional) Product Extensions
- (optional) PriceAnalyzer – data in Datamart

Then you need to create:

1. Calculation logic
2. (optional) Workflow logic
3. Price List (using the new calculation logic)

How to Configure a Calculated Field Set

To configure a Calculated Field Set:

1. Create Calculation Logic (Configuration > Calculation Logic > Generic Logic).
2. Create a Calculated Field Set (Configuration > Calculated Field Sets).
    a. Set a table which you want to iterate.
    b. Set a new logic which calculates the results.
    c. Map the destination fields of the table.

How to Build a Live Price Grid

To create a Live Price Grid (LPG):

1. Prepare a calculation logic with at least one element (whose Display Mode == Price Grid, otherwise you will not be able to pick up any Output fields in the Configuration of LPG).
2. Make a new LPG.
3. In a list of LPGs, select your LPG and using the action icons below the table do the following:
    a. Add products into the selected LPG.
        ℹ You must add some products, otherwise you will not be able to configure the LPG in the next step.
    b. Configure the LPG (i.e. add a logic into it, select Output Elements).
    c. Calculate it.

For details on the settings see Create and Configure New LPG.

How to Configure a Quote

As an administrator you want to configure a new quote calculation logic that will be used by sales representatives (end users) when creating new sales quotes.

As prerequisites you need:

- Product master – A quote is calculated for specific products.
- Customer master – A quote is calculated for a specified customer.
- Price list, price parameters, pricing extensions – The source of data for the price calculation.
- Optionally, you can prepare code for a chart in PriceAnalyzer.

When you have these items ready, you will create a calculation logic for the quote and a workflow for the quote approval:

1. Create a calculation logic for your quote lines. Decide what the user's input will be and what will be calculated by the logic. Use price lists, product extensions and/or price parameters as the data sources.
   Make sure this logic is set as the 'Default pricing logic' on the Calculation Logic overview page (Configuration > Calculation Logic > Generic Logic). Only the default pricing logic is used for calculating a new quote.
2. (Optional) Create a Quote Header Logic – for details see Quote Header Logics. This is used in special cases, when you need to perform an operation on the whole quote (over all the line items at once), for example, to classify a quote size, or calculate an average of the line price. See also an example.
3. (Optional) Define a logic that will restrict products available in the Product picker when adding products to a quote. (Sample usage: based on the selected customer, users will be able to select only a certain set of products.)
4. Create a quote to verify that the calculation logic works correctly.
5. (Optional) Set up columns of Price Records. These records are used to export the quote to external systems (e.g., ERP). They are generated automatically after the quote is converted from "Offer" to "Deal". The mapping from quote to PR is done via names; the name of the quote logic element must be the same as the name of the PR column. Also, the values entered by the user into the input fields (via UserEntry) will be copied to the PR columns (mapping by name, i.e. the name of the UserEntry must be the same as the name of the attribute).
6. Create a workflow for the approval process.

Quote Header Functions

To set up the logic for quote header, go to Quote Header Logics in Configuration.

You will work with the `quoteProcessor` variable and use the function `addOrUpdateOutput()`.

```
quoteProcessor.addOrUpdateOutput(
   "ROOT",
   [
     "resultName" : "totalinvoiceprice",
     "resultLabel": "Total Invoice Price",
     "result"     : "100 GBP"
   ]
)
```

It is important to understand that the header logic is executed twice: *before* the line item calculation and *after* it. The header logic can determine the phase it is in by `quoteProcessor.isPrePhase()` or `isPostPhase()`.

The reason to split this up is that certain actions should be performed before the lines are calculated (like input normalization, enhancement or rearranging lines). On the other hand, rollup and summary calculations can only be performed after the lines have been calculated.

Once the header logic has been executed, the resulting actions (data alteration actions are not executed immediately!) are performed in this sequence per every phase:

1. Add line items
2. Delete line items (only during pre-process, otherwise the rollups would be affected)
3. Move line items (only during pre-process, otherwise the rollups would be affected)
4. Rename folders
5. Add inputs
6. Add outputs
7. Update header fields (only some fields are supported now)

EXAMPLE

We have a quote with two lines.

When we run this code:

```
def qp = quoteProcessor;
def q = qp.getQuoteView();
def h = qp.getHelper();
for (lineItemMap in q.lineItems) {


  /* lineItem = h.lineItemFromMap(lineItemMap) */

  api.trace("lineItem","",lineItemMap);
}
```

Then we get in the trace:

```
{
  version=0,
  inputs=[
    {
      name=NettPriceGBP/km,
      label=NettPriceGBP/km,
      type=USERENTRY,
      parameterConfig={
        dataType=float
      }
    }
  ],
  outputs=[

  ],
  dirty=false,
  lineId=7xYQKCWFWckEaV3,
  calculationStatus=0,
  editabilityStatus=0,
  label=2x1.5FP400BLK,
  sku=20145871,
  folder=false,
  treeLabel=2x1.5FP400BLK,
  typedId=50205.QLI,
  createDate=2016-04-18T14: 57: 29,
  lastUpdateDate=2016-04-18T14: 57: 29
}
```

How to Configure PromotionManager

This example shows how to implement a basic project in Promotion Manager. We will create a promotional campaign, in which we want to offer to restaurants a volume discount on lamb meat and a net fixed price for all lamb meatballs.

To build a new project in PromotionManager, we will take the following steps:

1. Create a contract calculation logic for each of the two line items in the contract.
2. Create a contract term type for each of the calculation logics.
3. Set up Price Records – map contract attributes to Price Record fields.
4. Create a calculation logic for accessing the Price Records.
5. Typically, you will also to set up an approval workflow logic in Workflow Logics (of Contract type).

Create Contract Calculation Logics

We will create two contract calculation logics that define user inputs and pass the entered values on to Price Records. To start, go to Configuration > Calculation Logic > Contracts and click 'Add'.

VOLUME DISCOUNT CALCULATION LOGIC

The volume discount logic will include the following elements:

- User input field for selecting a customer group.
- User input field for selecting a product group.
- Contract term type – returns the reference to the contract term type of the line.
- User input field for selecting the start and end date of the campaign (if you want to give the user the option to modify the dates inherited from contract level).
- A multi-tier user input field for entering the volume discount ranges.

Element Name	Label	Formula Expression
CustomerGroup	Customer Group	`api.customerGroupEntry()`
ProductGroup	Product Group	`api.productGroupEntry()`

ContractTermType	Contract Term Type	`def nameFilter = Filter.equal("uniqueName", api.currentItem()?.contractTermType)`
		`def ctt = api.find("CTT", nameFilter)`
		`return ctt?.getAt(0)`

ValidFrom	Valid From	`def dateString = api.dateUserEntry("ValidFrom")`
		`def date = api.parseDate("yyyy-MM-dd",dateString)`
		`api.currentItem()?.startDate = date //we need to set the override here to be copied to the Price Record`
		`return date`

ValidTo	Valid To	`def dateString = api.dateUserEntry("ValidTo")`
		`def date = api.parseDate("yyyy-MM-dd", dateString)`
		`api.currentItem()?.endDate = date //we need to set the override here to be copied to Price Record`
		`return date`

VolumeDiscount	Volume Discount	`def volumeDiscountEntry = api.multiTierEntry("VolumeDiscount", "Qty", "%")`
		`return volumeDiscountEntry?.asMap()`

NET PRICE CALCULATION LOGIC

The net price logic will include the following elements:

- User input field for selecting a customer group.
- User input field for selecting a product group.
- Contract term type – returns the reference to the contract term type of the line.
- User input field for selecting the start and end date of the campaign (if you want to give the user the option to modify the dates inherited from contract level).
- User input field for the net price value.

Element Name	Label	Formula Expression
CustomerGroup	Customer Group	`api.customerGroupEntry()`

ProductGroup	Product Group	api.productGroupEntry()

ContractTermType	Contract Term Type	def nameFilter = Filter.equal("uniqueName", api.currentItem()?.contractTermType)
		def ctt = api.find("CTT", nameFilter)
		return ctt?.getAt(0)

ValidFrom	Valid From	def dateString = api.dateUserEntry("ValidFrom")
		def date = api.parseDate("yyyy-MM-dd",dateString)
		api.currentItem()?.startDate = date //we need to set the override here to be copied to the Price Record
		return date

ValidTo	Valid To	def dateString = api.dateUserEntry("ValidTo")
		def date = api.parseDate("yyyy-MM-dd", dateString)
		api.currentItem()?.endDate = date //we need to set the override here to be copied to Price Record
		return date

NetPrice	Net Price	return api.userEntry("Net Price")?:0

Create Contract Term Types

A Contract term type represents an item in the promotional offer. We will create two contract term types, one for the volume discount and one for the net price promotion. We will assign the previously created calculation logics to them.

1. Go to 'PromotionManager' > 'Contract Term Types'.
2. Click 'Add' and enter the name and label.
3. Select the corresponding Pricing logic.
4. (Optional) Define the value of the Waterfall Element and other attributes.
   In the Waterfall Element, you can indicate which field (of the Quote or Datamart) is influenced by this contract condition. This can be used, for example, when the Quote Line searches for the appropriate Price Record (which was generated from the Contract Term Type).

Set Up Price Records

Price Records are created automatically from the Contract lines but you can influence the content of the resulting fields. In our case most of the things go automatically, we only need an additional "VolumeDiscount" element.

Source	Destination Price Record	
	Field Name	Field Label
`currentItem().startDate`	validAfter	
`currentItem().endDate`	expiryDate	
Logic element named "VolumeDiscount"		VolumeDiscount

For more details see Price Records Generation from Contract.

Create Logic to Access the Price Records

This example shows how to find an appropriate Price Record on the Quote Line Item.

```
def productFilter = api.productToRelatedObjectsFilter("PR",
api.product("sku"))
def customerFilter = api.customerToRelatedObjectsFilter("PR",
api.customer("customerId"))
def dateFromFilter = Filter.lessOrEqual("validAfter",api.targetDate())
def dateToFilter = Filter.greaterOrEqual("expiryDate",api.targetDate())

def prs = api.find("PR", productFilter, customerFilter, dateFromFilter,
dateToFilter)

for (pr in prs) {
 //...........
}
```

See also Searching Relevant Price Records.

How to Configure RebateManager

To configure the RebateManager, so that you can create new rebate agreements, take the following steps:

1. Make sure you have the following prerequisites ready:
    - Customer master
    - Product master
    - Datamart
2. Create a Rebate calculation logic.
3. Create a How to Configure RebateManager#Rebate Type.
4. Create a How to Configure RebateManager#Rebate Agreement.
5. Create How to Configure RebateManager#Rebate Records.
6. Create a workflow for rebate and payout approval.
7. Create an allocation calculation logic for allocating data to a datamart.

Rebate Calculation Logic

Rebate calculation logic is used to calculate:

- Rebate Agreement Line
- Rebate Agreement Line (Read-only) – for approved Agreements
- Rebate Record

Because the logic is almost the same for all of these objects, you define only one logic, and if an element is specific only to some of these objects, you set the "Group" of the element.

Element Name	Calculation Context	Groovy	Note
ProductGroup		`api.productGroupEntry()`	The conditions are typically negotiated for a group of products.
CustomerGroup		`api.customerGroupEntry()`	The conditions are typically negotiated for a group of customers.
RebateDiscount		`api.userEntry("RebateDiscount") / 100.0`	The negotiated condition – in this case the discount in percentage.
AbortOnSyntaxCheck		`if (api.isSyntaxCheck()) api.abortCalculation()`	
Today		`return api.targetDate()`	This is to be able to simulate different "today"
StartDate		`return (api.currentItem()?.startDate) ? api.parseDate("yyyy-MM-dd",api.currentItem().startDate) : new Date()`	

<table>
<tr><td>EndDate</td><td></td><td><pre>return (api.currentItem()?.endDate) ?
api.parseDate("yyyy-MM-dd",api.currentItem().endDate) : new Date()</pre></td><td></td></tr>
<tr><td>EndDateOrToday</td><td></td><td><pre>return (api.getElement("Today") < api.getElement("EndDate")) ?
api.getElement("Today") : api.getElement("EndDate")</pre></td><td></td></tr>
<tr><td>ActualSales</td><td></td><td><pre>def dmCtx = api.getDatamartContext()
def salesDM = dmCtx.getDatamart("Transactions")
def datamartQuery = dmCtx.newQuery(salesDM,true)
datamartQuery.select("SUM(Invoice_Price)", "IP")
datamartQuery.where(

Filter.greaterOrEqual("Invoice_Date",api.getElement("StartDate")),

Filter.lessOrEqual("Invoice_Date",api.getElement("EndDateOrToday"))
)
datamartQuery.where(api.getElement("CustomerGroup"))
datamartQuery.where(api.getElement("ProductGroup"))
def result = dmCtx.executeQuery(datamartQuery)
return (result?.getData()?.getValue(0,0)?:0.0)</pre></td><td>How much money were actually spent in the time frame. The number comes from Datamart for the specified period.</td></tr>
<tr><td>Rebate</td><td></td><td><pre>def sales = api.getElement("ActualSales")
def rebatePct = api.getElement("RebateDiscount")

return (sales != null && rebatePct != null) ? (sales * rebatePct)
: null</pre></td><td>SimpleRebate calculation – we will pay back the customer the negotiated percentage of the spendings on the given group of products.</td></tr>
<tr><td>ForecastSales</td><td></td><td><pre>def sales = api.getElement("ActualSales")
if (sales<=0) sales=0

if (sales > 0) {
    def from = api.getElement("StartDate")
    def to = api.getElement("EndDate")
    def today = api.targetDate()

    if (today > to) return sales
    else if (today < from) return 0.0
    else {
        int totalDays = to - from + 1
        def daysofTransactions = today - from + 1
        def perDaySales = sales / daysofTransactions

        //api.trace("validity", from as String, to as String)
        //api.trace("totalDays", totalDays)
        //api.trace("daysofTransactions",daysofTransactions)

        return perDaySales * totalDays
    }
}</pre></td><td></td></tr>
<tr><td>ForecastRebate</td><td></td><td><pre>def sales = api.getElement("ForecastSales")
def rebatePct = api.getElement("Rebate")

return (sales != null && rebatePct != null) ? (sales * rebatePct)
: null</pre></td><td></td></tr>
<tr><td>RebateRecordCreate</td><td>Agreement</td><td><pre>rebateRecords.add()

/*

rebateRecords.add([
"attribute1":api.getElement("Rebate"),
"attribute2":api.getElement("TotalSales"),
"attribute3":api.getElement("RebateDiscount")
])

*/</pre></td><td>Every agreement line will generate one Rebate Record (i.e. there will be only one payout at the end of the year).</td></tr>
<tr><td>RRInput</td><td>RebateRecord</td><td><pre>api.userEntry("RRInput")</pre></td><td>This is an example of how to make an input specific for the Rebate Record.</td></tr>
</table>

CalculationBase	RebateRecord	`calculationBase.clear()`
		`calculationBase.include(api.getElement("CustomerGroup"))`
		`calculationBase.include(api.getElement("ProductGroup"))`
		`calculationBase.include(`
		`   new TimePeriod(api.getElement("StartDate"),`
		`api.getElement("EndDateOrToday"), TimeUnit.DAY)`
		`)`
		`calculationBase.setDateFieldName("InvoiceDate")`
			Calculation base specifies a filter which finds the sales transactions related to this Rebate Record.

Rebate Type

You need to create a new Rebate Type:

Label	Pricing Logic	Waterfall Element
Rebate Bonus	<your Rebate Calculation Logic>	This OOTB column is here only in case you would like to use it. Here you can store the name of the waterfall element which is influenced by this Rebate.

Rebate Agreement

Now you can create a new Rebate Agreement and add a new line using the new Rebate Type. See Add/Edit Rebate Agreements for details.

Rebate Records

Rebate Records are calculations of how much rebate will be paid out to customer, rebate estimations, etc. One Rebate Record represents one payout.

Rebate Records are:

- Calculated after the agreement validity period to find out how much rebate will be paid.
- Submitted for approval before the money is actually paid out to the customer.
- Sent to SAP to process the payment.

The record has over 100 attributes that can be used.

For more information, see Rebate Records.

Rebate Record Calculation Results Mapping to the Attributes

After you calculate the RR and Save it, the Calculation Result can be mapped to the Attributes of the Rebate Record.

The copy process happens automatically, when you save the Rebate Record.

The mapping rules are the same as for Price Record, i.e. when the name of the Element of the Rebate Type Logic is the same as the name of the attribute column (or the RR), the system will copy the element result value into the attribute.

Rebate Allocation

If you want to allocate the rebate money (paid to the customer) back to the sales transactions you need to create:

- Feeder Logic (in PriceBuilder)
- Allocation Logic (in PriceAnalyzer > Data Manager)
- Data Load

FEEDER LOGIC (IN PRICEBUILDER)

Type	Element Name	Expression
Formula	StartDate	`DateUserEntry("StartDate")`
Formula	EndDate	`DateUserEntry("EndDate")`

Groovy	EmitRebateRecords	`if(api.isSyntaxCheck())return`
		
		`def startDate = api.getElement("StartDate")`
		`def endDate = api.getElement("EndDate")`
		
		`if(startDate == null || endDate == null){`
		`def cal = api.datamartCalendar()`
		`def year = cal.getTimePeriod(cal.getYear(api.targetDate()))`
		`startDate = year.getStartDate()`
		`endDate = year.getEndDate()`
		
		`}`
		
		`def filter = Filter.and(`
		`Filter.greaterOrEqual("startDate", startDate),`
		`Filter.lessOrEqual("endDate",endDate)`
		`)`
		
		`def dateFieldName = "Invoice_Date"`
		`def sortRRByField = "startDate"`
		`api.emitRebateRecords(dateFieldName, sortRRByField, filter)`

ALLOCATION LOGIC (IN PRICEANALYZER > DATA MANAGER)

Type	Element Name	Display Mode	Groups	Expression	Description
Formula	InvoicePrice	Never	Row	`UserEntry("InvoicePrice")`	Invoice price of the transaction (to which we allocate the rebate).
Formula	PreviousRebate	Never	Row	`UserEntry("PrevRebate")`	Actual Rebate value on the transaction (to which we allocate the rebate).
Formula	Rebate	Everywhere	Row	`PreviousRebate + (InvoicePrice * CurrentItem("attribute1") / CurrentItem("attribute3"))`	The result Rebate value, which should be on the Transaction after adding the Rebate from the Rebate Record. The CurrentItem() returns the Rebate Record being allocated. We take only proportional part of the Rebate. attribute1 is the Rebate value of the Rebate Record to be allocated. attribute3 is the Total Sales (i.e. summary of Invoice Prices of all the transactions which were used to calculate the Rebate).

DATA LOAD

Create a new Calculation Data Load:

- Formula – Your new allocation PA logic.
- Feeder – Your new Feeder logic.
  - Input Start Date – Typically set to the beginning of the year, where you want to allocate.
  - Input End Date – Typically set to the end of the year, where you want to allocate.
- Formula Input (Source Field Mapping)
  - InvoicePrice – Transaction column "Invoice Price".
  - PreviousRebate – Transaction column "Rebate".
- Formula Outputs (Target Fields)
  - Rebate – The output mapping is done in the way that the element name must equal the column name (of the transaction).

Before you run the allocation data load, you need to have some Rebate Records for that period, otherwise not much happens.

RebateManager Calculation Logic Examples

In this section we will show four examples of logics that calculate different types of rebates:

- Tiered rebate – based on quantities or amounts purchased.
- Stepped rebate – tiered rebate where the rebate is calculated separately for each tier.

- Growth rebate – based on a specified increase in purchases.
- Fixed rebate – not dependent on the purchased amounts.

See also the Rebate Types section below for detailed explanation of the rebate types.

TIERED REBATE

The complete calculation logic will have the following elements:

Type	Element Name	Display Mode	Format Type
{}	Library	Never	
{}	RetainGlobal	Never	
{}	CustomerGroup	Never	
{}	ProductGroup	Never	
{}	SalesTarget	Never	
{}	ActualsQuery	Never	
{}	ActualSales	Never	
{}	TotalSalesRebate	QuoteConfigurator	Money (EUR)
+−÷%	ASFFactor	Never	
+−÷%	SalesForecast	QuoteConfigurator	Money (EUR)
{}	RebateForecast	QuoteConfigurator	Money (EUR)
{}	RebateRecords	Never	

Library – Stores the actual calculation of the rebate.

Library

```
// Tiered discount calculation function: when revenue falls in specific
tier, %discount of that tier is applied.
// If the base for the discount calculation is different from the
actual, provide a discountBase argument.
// Target values are % values (i.e. 5% is represented by 5 rather than
0.05)
def calcDiscount(actual, tieredTarget){
  return calcDiscount(actual, tieredTarget, null)
}

def calcDiscount(actual, tieredTarget, discountBase){
  api.log("calcDiscount: actual", actual.inspect())
  api.log("calcDiscount: target", tieredTarget.inspect())

  if (actual != null && tieredTarget != null){
    if (tieredTarget.size()>0){
      for (i = tieredTarget.size()-1; i>=0; i--){

        def tier = tieredTarget.get(i)
        def target = tier.target as BigDecimal
        def value = tier.value as BigDecimal
        api.log("Tier" +i + " target", target.inspect())
        api.log("Tier" +i + " value", value.inspect())

        if (target != null){
          if (actual > target){
            if (value == null) value=0
            def discount = (discountBase == null? actual : discountBase)
* value
            api.log("  Discount", discount)
            return discount
          }
        }
      }
    }
  }
  return 0.0
}
```

RetainGlobal – The usage of global variables must be explicitly enabled.

✓ From 3.8 Manhattan, api.retainGlobal can be set to true by default in Configuration.

RetainGlobal

```
api.retainGlobal=true
```

CustomerGroup user input – Displays a field where users select the customer group for which the rebate is calculated.

CustomerGroup

```
def cg = api.customerGroupEntry()
api.log("customer group", cg.inspect())

return cg
```

ProductGroup user entry – Displays a field where users select the product group for which the rebate is calculated.

ProductGroup

```
def pg = api.productGroupEntry()
api.log("product group", pg.inspect())

return pg
```

SalesTarget user entry – Displays a field where users can define the number of tiers and their thresholds.

SalesTarget

```
def tieredVal = api.multiTierEntry("Revenue Target", "€","%")
if (api.isSyntaxCheck()) return
// 5% --> 0.05 conversion
if (tieredVal != null){
  tieredVal = tieredVal.multiplyValues(0.01)
}
api.log("sales target", tieredVal.inspect())
api.global.target = tieredVal
return tieredVal
```

Actuals – Datamart Query

ActualsQuery

```
if (api.isSyntaxCheck()) return
def line = api.currentItem()
if (line != null){  // is null on syntax check/formula save
  def customerGroup = api.getElement("CustomerGroup")
  def productGroup = api.getElement("ProductGroup")

  // actuals for the period specified in the rebate line item
  def timeFilter = api.filter("InvoiceDate", line?.startDate,
line?.endDate)
  if (timeFilter != null){
   api.datamartQuery("TX", "Transaction DM", customerGroup,
productGroup, timeFilter, "InvoicePrice\$")
  }
}
```

ActualSales – Datamart Lookup

ActualSales

```
if (api.isSyntaxCheck()) return
def sales = api.datamartLookup("TX", "InvoicePrice\$")
api.log("actual", sales.inspect())

api.global.actual = sales as BigDecimal
return api.global.actual
```

TotalSalesRebate – Calculation of the rebate.

TotalSalesRebate

```
if (api.isSyntaxCheck()) return
def sales = api.global.actual
if (sales<0) sales=0

def rebate = Library.calcDiscount(sales, api.global.target)
api.global.rebate = rebate as BigDecimal
return api.global.rebate
```

ASF Factor – The expected growth of sales (example).

ASFFactor

```
1.75
```

SalesForecast – Actual sales multiplied by the expected growth rate.

SalesForecast

```
ActualSales*ASFFactor
```

RebateForecast – Expected rebate based on the forecasted sales.

RebateForecast

```
if (api.isSyntaxCheck()) return
def sales = api.global.actual * api.getElement("ASFFactor")
if (sales<0) sales=0

def rebate = Library.calcDiscount(sales, api.global.target)
api.global.rebate = rebate as BigDecimal
return api.global.rebate
```

RebateRecords – Creates a new rebate record for the calculated rebate.

RebateRecords

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

STEPPED REBATE

The complete calculation logic will have the following elements:

Type	Element Name	Display Mode	Format Type
{}	Library	Never	
{}	RetainGlobal	Never	
{}	CustomerGroup	Never	
{}	ProductGroup	Never	
{}	SalesTarget	Never	

{}	ActualsQuery	Never	
{}	ActualSales	Never	
{}	TotalSalesRebate	QuoteConfigurator	Money (EUR)
+ − ÷ %	ASFFactor	Never	
+ − ÷ %	SalesForecast	QuoteConfigurator	Money (EUR)
{}	RebateForecast	QuoteConfigurator	Money (EUR)
{}	RebateRecords	Never	

Library – Stores the actual calculation of the rebate.

Library

```
//Stepped discount calculation function: revenue falls in a specific tier, %discount of each tier up to this tier is applied to the part of the revenue.
if (api.isSyntaxCheck()) return

// target values are % values (i.e. 5% is represented by 5 rather than 0.05)
def calcSteppedDiscount(actual, tieredTarget){
  api.log("calcSteppedDiscount: actual", actual.inspect())
  api.log("calcSteppedDiscount: target", tieredTarget.inspect())

  def cumulativeDiscount = 0.0
  if (actual != null && tieredTarget != null){
    if (tieredTarget.size()>0){
      def unusedActual = actual
      for (i = tieredTarget.size()-1; i>=0; i--){
       def tier = tieredTarget.get(i)
       def target = tier.target as BigDecimal
       def value = tier.value as BigDecimal
       api.log("Tier" +i + " target", target.inspect())
       api.log("Tier" +i + " value", value.inspect())

       if (target != null){
          if (unusedActual > target){
           if (value == null) value=0

           def discountable = unusedActual - target
           def discount = discountable * value
            api.log("Discountable", discountable)
            api.log("Discount", discount)

            cumulativeDiscount += (discountable * value)
            api.log("  Cumulative discount", cumulativeDiscount)
            unusedActual = target
          }
        }
      }
    }
  }
  return cumulativeDiscount
}
```

RetainGlobal – The usage of global variables must be explicitly enabled.

✓ From 3.8 Manhattan, api.retainGlobal can be set to true by default in Configuration.

RetainGlobal

```
api.retainGlobal=true
```

CustomerGroup user input – Displays a field where users select the customer group for which the rebate is calculated.

CustomerGroup

```
def cg = api.customerGroupEntry()
api.log("customer group", cg.inspect())

return cg
```

ProductGroup user entry – Displays a field where users select the product group for which the rebate is calculated.

ProductGroup

```
def pg = api.productGroupEntry()
api.log("product group", pg.inspect())

return pg
```

SalesTarget user entry – Displays a field where users can define the number of tiers and their thresholds.

SalesTarget

```
def tieredVal = api.multiTierEntry("Revenue Target", "€","%")
if (api.isSyntaxCheck()) return
// 5% --> 0.05 conversion
if (tieredVal != null){
  tieredVal = tieredVal.multiplyValues(0.01)
}
api.log("sales target", tieredVal.inspect())
api.global.target = tieredVal
return tieredVal
```

Actuals – Datamart Query

ActualsQuery

```
if (api.isSyntaxCheck()) return
def line = api.currentItem()
if (line != null){  // is null on syntax check/formula save
  def customerGroup = api.getElement("CustomerGroup")
  def productGroup = api.getElement("ProductGroup")

  // actuals for the period specified in the rebate line item
  def timeFilter = api.filter("InvoiceDate", line?.startDate,
line?.endDate)
  if (timeFilter != null){
   api.datamartQuery("TX", "Transaction DM", customerGroup,
productGroup, timeFilter, "InvoicePrice\$")
  }
}
```

ActualSales – Datamart Lookup

ActualSales

```
if (api.isSyntaxCheck()) return
def sales = api.datamartLookup("TX", "InvoicePrice\$")
api.log("actual", sales.inspect())

api.global.actual = sales as BigDecimal
return api.global.actual
```

TotalSalesRebate – Calculation of the rebate.

TotalSalesRebate

```
if (api.isSyntaxCheck()) return
def sales = api.global.actual
if (sales<0) sales=0

def rebate = Library.calcSteppedDiscount(sales, api.global.target)
api.global.rebate = rebate as BigDecimal
return api.global.rebate
```

ASF Factor – The expected growth of sales.

ASFFactor

```
1.75
```

SalesForecast – Actual sales multiplied by the expected growth rate.

SalesForecast

```
ActualSales*ASFFactor
```

RebateForecast – Expected rebate based on the forecasted sales.

RebateForecast

```
if (api.isSyntaxCheck()) return
def sales = api.global.actual * api.getElement("ASFFactor")
if (sales<0) sales=0

def rebate = Library.calcDiscount(sales, api.global.target)
api.global.rebate = rebate as BigDecimal
return api.global.rebate
```

RebateRecords – Creates a new rebate record for the calculated rebate.

RebateRecords

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

GROWTH REBATE

The complete calculation logic will have the following elements:

Type	Element Name	Display Mode	Format Type
{}	Library	Never	
{}	RetainGlobal	Never	
{}	CustomerGroup	Never	
{}	ProductGroup	Never	
{}	GrowthTarget	Never	

{}	ActualsQuery	Never	
{}	ActualSales	Never	
{}	PreviousYearSales	Never	
{}	GrowthRebate	QuoteConfigurator	Money (EUR)
+−×%	ASFFactor	Never	
+−×%	SalesForecast	QuoteConfigurator	Money (EUR)
{}	RebateForecast	QuoteConfigurator	Money (EUR)
{}	RebateRecords	Never	

Library – Stores the actual calculation of the rebate.

Library

```
//When increase to the pervious period's actual is higher than a given target percentage
//target levels and values are % values (i.e. 5% is represented by 5 rather than 0.05)
def growthBonus(actual, prevPeriodActual, tieredTarget){
  api.log("growthBonus: actual", actual.inspect())
  api.log("growthBonus: prevPeriodactual", prevPeriodActual.inspect())
  api.log("growthBonus: target", tieredTarget.inspect())

  if (actual != null && prevPeriodActual != null && prevPeriodActual != 0.0 && tieredTarget != null){
   actual = actual as BigDecimal  // to be safe (DatamartQuery can return doubles)
   prevPeriodActual = prevPeriodActual as BigDecimal
    def growth = 100* (actual - prevPeriodActual) / prevPeriodActual  // as %
    api.log("growthBonus: growth", growth.inspect())
    return calcDiscount(growth, tieredTarget, actual)
  }
}
```

RetainGlobal – The usage of global variables must be explicitly enabled.

✓ From 3.8 Manhattan, api.retainGlobal can be set to true by default in Configuration.

RetainGlobal

```
api.retainGlobal=true
```

CustomerGroup user input – Displays a field where users select the customer group for which the rebate is calculated.

CustomerGroup

```
def cg = api.customerGroupEntry()
api.log("customer group", cg.inspect())

return cg
```

ProductGroup user entry – Displays a field where users select the product group for which the rebate is calculated.

ProductGroup

```
def pg = api.productGroupEntry()
api.log("product group", pg.inspect())

return pg
```

GrowthTarget user entry – Displays a field where users can define the number of tiers and their thresholds.

GrowthTarget

```
def tieredVal = api.multiTierEntry("Growth Target", "%","%")
if (api.isSyntaxCheck()) return
//rebate % conversion
if (tieredVal != null){
  tieredVal = tieredVal.multiplyValues(0.01)
}
api.log("growth target", tieredVal.inspect())
api.global.target = tieredVal
return tieredVal
```

ActualsQuery – Datamart query to find out actual and last year's sales.

ActualsQuery

```
if (api.isSyntaxCheck()) return
def line = api.currentItem()
if (line != null){  // is null on syntax check/formula save
  def customerGroup = api.getElement("CustomerGroup")
  def productGroup = api.getElement("ProductGroup")

  // actuals for the period specified in the rebate line item
  def timeFilter = api.filter("InvoiceDate", line?.startDate,
line?.endDate)
  if (timeFilter != null){
   api.datamartQuery("TX", "Transaction DM", customerGroup,
productGroup, timeFilter, "InvoicePrice\$")
  }

  // previous year (copy and paste for other past periods)
  def cal = api.datamartCalendar()

  def startDate = line?.startDate?.toString()  // is a string
  def endDate = line.endDate.toString()  // is a string

  if (startDate != null){
 startDate = cal.add(startDate, -1, "YEAR");
  }
  if (endDate != null){
    endDate = cal.add(endDate, -1, "YEAR");
  }
  timeFilter = api.filter("InvoiceDate", startDate, endDate)
  if (timeFilter != null){
    api.datamartQuery("TX-1yr", "Transaction DM", customerGroup,
productGroup, timeFilter, "InvoicePrice\$")
  }
}
```

ActualSales – Datamart Lookup

ActualSales

```
if (api.isSyntaxCheck()) return
def sales = api.datamartLookup("TX", "InvoicePrice\$")
api.log("actual", sales.inspect())

api.global.actual = sales as BigDecimal
return api.global.actual
```

PrevYearSales – Datamart Lookup

PrevYearSales

```
if (api.isSyntaxCheck()) return
def sales = api.datamartLookup("TX-1yr", "InvoicePrice\$")
api.log("actual", sales.inspect())

api.global.actualPrevYear = sales as BigDecimal
return api.global.actualPrevYear
```

GrowthRebate – Calculation of the rebate based on the sales growth.

GrowthRebate

```
if (api.isSyntaxCheck()) return
def sales = api.global.actual
def lastYearsSales = api.global.actualPrevYear

if (sales<0) sales=0
if (lastYearsSales<0) lastYearsSales=0

def rebate = Library.growthBonus(sales, lastYearsSales,
api.global.target)
api.global.rebate = rebate as BigDecimal
return api.global.rebate
```

ASF Factor – The expected growth of sales.

ASFFactor

```
1.75
```

SalesForecast – Actual sales multiplied by the expected growth rate.

SalesForecast

```
ActualSales*ASFFactor
```

RebateForecast – Expected rebate based on the forecasted sales.

RebateForecast

```
if (api.isSyntaxCheck()) return
def sales = api.global.actual * api.getElement("ASFFactor")
if (sales<0) sales=0

def rebate = Library.calcDiscount(sales, api.global.target)
api.global.rebate = rebate as BigDecimal
return api.global.rebate
```

RebateRecords – Creates a new rebate record for the calculated rebate.

RebateRecords

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

FIXED REBATE

The complete calculation logic will have the following elements:

Type	Element Name	Display Mode	Format Type
{}	CustomerGroup	Never	
{}	ProductGroup	Never	
+− +%	FixedRebate	QuoteConfigurator	Money (EUR)
{}	RebateRecords	Never	

CustomerGroup user input – Displays a field where users select the customer group for which the rebate is calculated.

CustomerGroup

```
def cg = api.customerGroupEntry()
api.log("customer group", cg.inspect())

return cg
```

ProductGroup user entry – Displays a field where users select the product group for which the rebate is calculated.

ProductGroup

```
def pg = api.productGroupEntry()
api.log("product group", pg.inspect())

return pg
```

FixedRebate user entry – Displays a field where the user can enter a fixed rebate value.

FixedRebate

```
UserEntry("Fixed rebate")
```

RebateRecords – Creates a new rebate record for the calculated rebate.

RebateRecords

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

REBATE VARIANTS

The following types are only examples, in Pricefx you can set up any rebate type you need.

Tiered Rebate

This is the simplest rebate type that is based on quantities or amounts purchased by the customer. The tiered rebate type defines several tiers with corresponding rebate percentages. The paid out rebate depends on the highest threshold level that has been reached.

Example: The final sales is $17,200 and the bonus is calculated based on three agreed tiers.

Threshold	Rebate Percentage	Threshold Reached	Calculation	Bonus
$10,000	1.0%	Yes		
$15,000	1.5%	Yes	17,200 * 0.015	$258
$20,000	2.0%	No		

Growth Rebate

The growth rebate type is based on a specified increase in purchases. Again, several tiers are defined and the paid out rebate depends on the threshold that has been reached.

Example: The previous year's sales was $14,000 and this year's sales has reached $17,200, so the increase is 22.86%. The bonus is calculated in the following way:

Threshold	Rebate Percentage	Threshold Reached	Calculation	Bonus
10%	1.0%	Yes		
20%	1.5%	Yes	17,200 * 0.015	$258
30%	2.0%	No		

Stepped Rebate

The bonus is paid for each tier whose lower threshold has been exceeded but only from the value that is within the tier limits and the percentage specified for that tier is applied.

Example: The final sales is $17,200. The first tier has been reached and exceeded, so the bonus is paid from the full interval, which is $5,000. The second tier has been reached but not exceeded, so the bonus is paid only from the value above the $15,000 threshold, i.e. $2,200.

Threshold	Rebate Percentage	Threshold Reached	Unused Actual	Discount Base	Calculation	Bonus
$10,000	1.0%	Yes		5,000	5,000 * 0.01	$50
$15,000	1.5%	Yes		2,200	2,200 * 0.015	$33
$20,000	2.0%	No				

The total paid out rebate is $83.

Rebate Agreement Template Simple Working Example

A simplistic example of a logic for Rebate Template. This logic creates one agreement with one line.

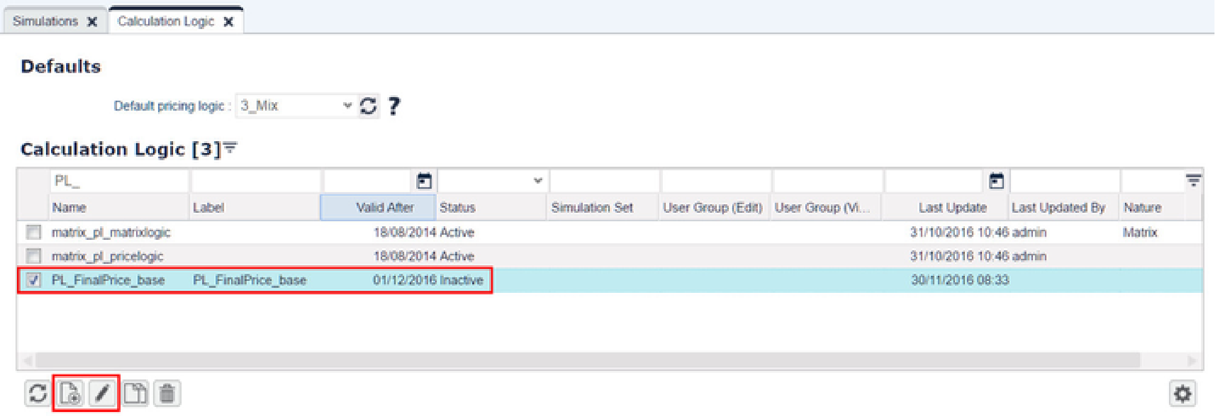
How to use:

1. Place the code to a new logic under Administration > Configuration > RebateManager > Rebate Agreement Template Logics. IMPORTANT: Review the first 3 lines and change it according to your system:
    a. rebateTypeName – Must be the name of an existing Rebate Type, otherwise it fails.
    b. customerId – Should be an existing customer Id, but from our experience it may also work with a non-existent Id (advisable for testing purposes only).
    c. inputFieldOnRebateType – The name of the input parameter on the line item. It must be the NAME of the input (in case you have the Name different from Label of the input parameter).
2. Make a new template record under RebateManager > Rebate Agreement Templates and assign the new logic to it.
3. Navigate to RebateManager > Rebate Agreements and click the "Create Rebate Agreements from Template" button.

How to Create a Price List Simulation

This example shows a how to create a simulation which uses a new pricing logic.

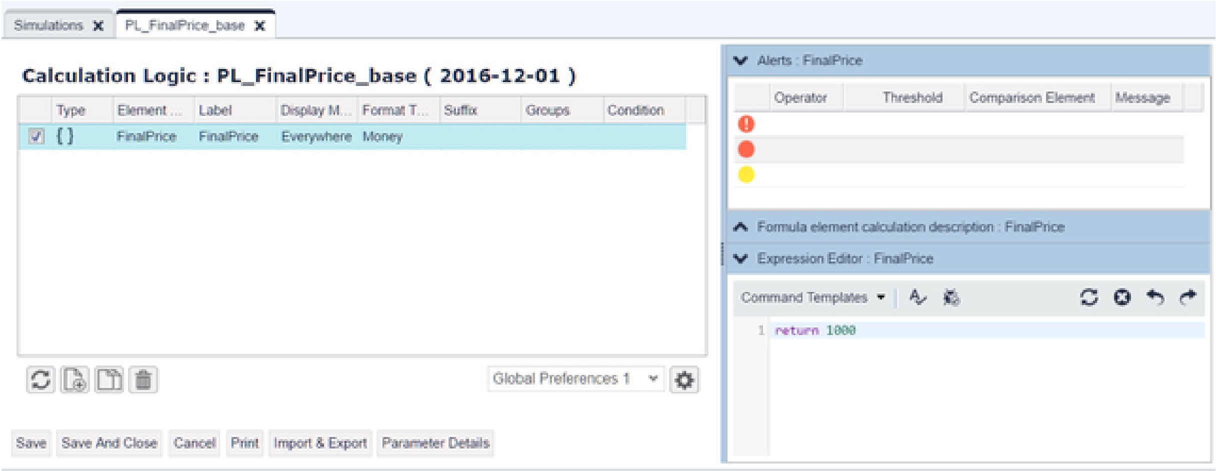
1. Create a pricing logic:
    a. In Configuration > Calculation Logic > Generic Logic, click the Add icon.
    b. Fill in Name, Label, Valid After and Status.
    c. Click the View/Edit icon.



PL_

	Name	Label	Valid After	Status	Simulation Set	User Group (Edit)	User Group (Vi...	Last Update	Last Updated By	Nature
□	matrix_pl_matrixlogic		18/08/2014	Active				31/10/2016 10:46	admin	Matrix
□	matrix_pl_pricelogic		18/08/2014	Active				31/10/2016 10:46	admin	
☑	PL_FinalPrice_base	PL_FinalPrice_base	01/12/2016	Inactive				30/11/2016 08:33		

2. Set up the calculation logic:
    a. Create a price element which will always return a value of 1000 (just as a test in this example).



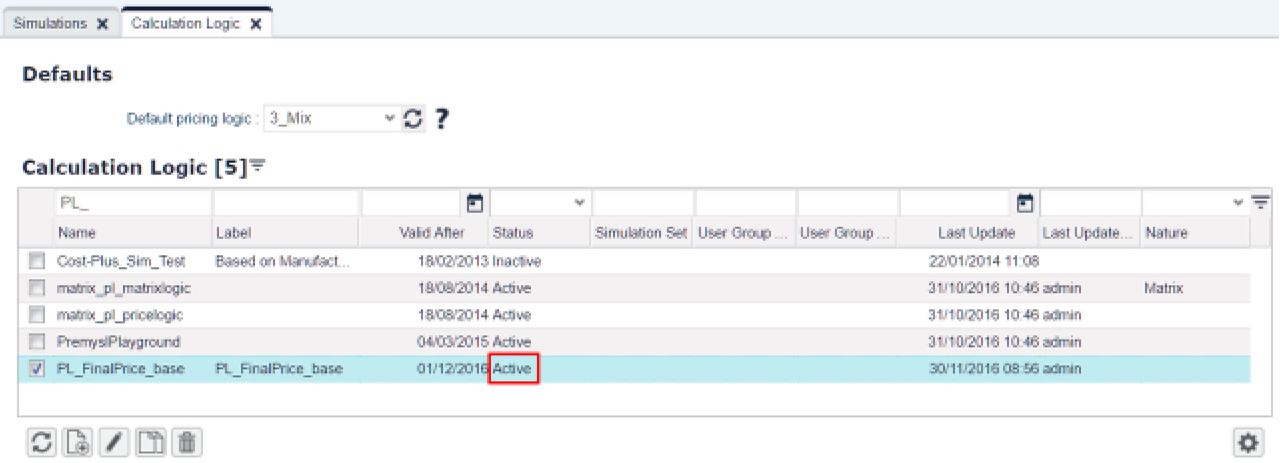
Calculation Logic : PL_FinalPrice_base (2016-12-01)

	Type	Element ...	Label	Display M...	Format T...	Suffix	Groups	Condition
☑	{ }	FinalPrice	FinalPrice	Everywhere	Money			

Operator	Threshold	Comparison Element	Message

```
return 1000
```

3. Activate the pricing logic.



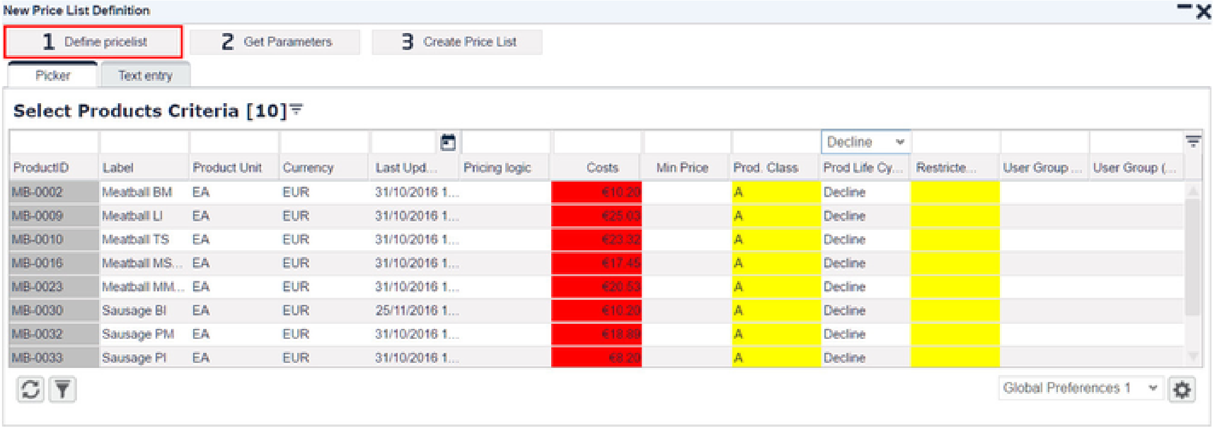
Defaults

Default pricing logic : 3_Mix

Calculation Logic [5]

	Name	Label	Valid After	Status	Simulation Set	User Group ...	User Group ...	Last Update	Last Update...	Nature
	PL_									
☐	Cost-Plus_Sim_Test	Based on Manufact...	18/02/2013	Inactive				22/01/2014 11:08		
☐	matrix_pl_matrixlogic		18/08/2014	Active				31/10/2016 10:46	admin	Matrix
☐	matrix_pl_pricelogic		18/08/2014	Active				31/10/2016 10:46	admin	
☐	PremyslPlayground		04/03/2015	Active				31/10/2016 10:46	admin	
☑	PL_FinalPrice_base	PL_FinalPrice_base	01/12/2016	Active				30/11/2016 08:56	admin	

4. Create a Price List:
    a. Go to PriceBuilder > Price Lists and click the Add icon.
    b. Select step 1 Define Price List.
    c. Select products for which you want to generate the price list. (You may want to use the filter, so that the list of products is not too long for the sample run.)



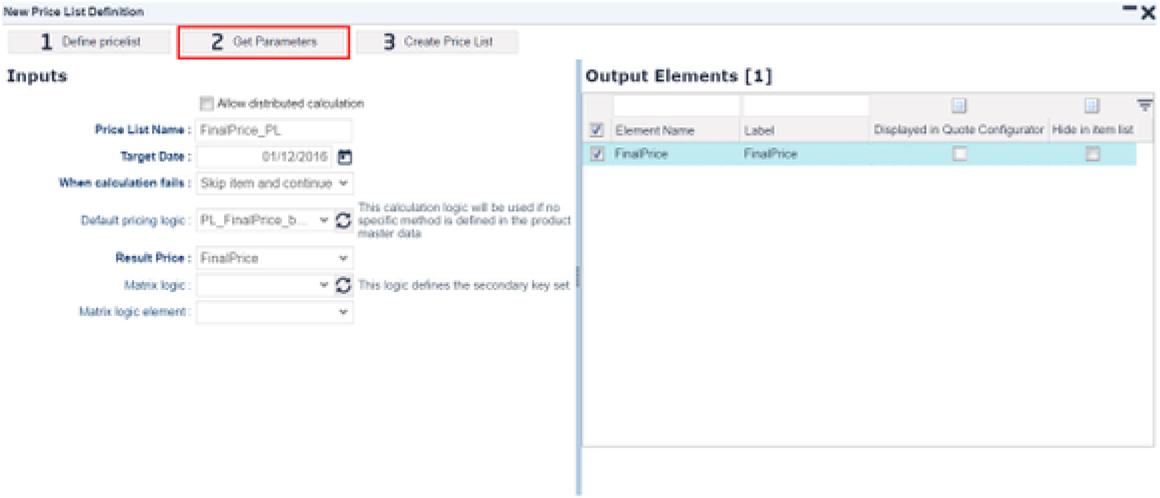
Select Products Criteria [10]

ProductID	Label	Product Unit	Currency	Last Upd...	Pricing logic	Costs	Min Price	Prod. Class	Prod Life Cy...	Restricte...	User Group ...	User Group (...
									Decline			
MB-0002	Meatball BM	EA	EUR	31/10/2016 1...		€10.20		A	Decline			
MB-0009	Meatball LI	EA	EUR	31/10/2016 1...		€25.03		A	Decline			
MB-0010	Meatball TS	EA	EUR	31/10/2016 1...		€23.32		A	Decline			
MB-0016	Meatball MS...	EA	EUR	31/10/2016 1...		€17.45		A	Decline			
MB-0023	Meatball MM...	EA	EUR	31/10/2016 1...		€20.53		A	Decline			
MB-0030	Sausage BI	EA	EUR	25/11/2016 1...		€10.20		A	Decline			
MB-0032	Sausage PM	EA	EUR	31/10/2016 1...		€18.89		A	Decline			
MB-0033	Sausage PI	EA	EUR	31/10/2016 1...		€8.20		A	Decline			

5. Associate the pricing logic with Price List.
    a. Click step 2 Get Parameters.
    b. Fill in Price List Name.
    c. Select Target Date – the date which will be used for looking up valid parameters. In this case we will take into account a pricing logic valid on Dec 1.
    d. Select your Pricing Logic. It must be Active.

e. Select your Result Price. It will show price elements from your selected pricing logic. Since our pricing logic (PL_FinalPrice_base) has only one element, it is the only one you can select.

f. In the Output Elements section, check the FinalPrice checkbox to make the element visible on your generated price list.



Inputs

☐ Allow distributed calculation

Price List Name : FinalPrice_PL

Target Date : 01/12/2016

When calculation fails : Skip item and continue

Default pricing logic : PL_FinalPrice_b... — This calculation logic will be used if no specific method is defined in the product master data

Result Price : FinalPrice

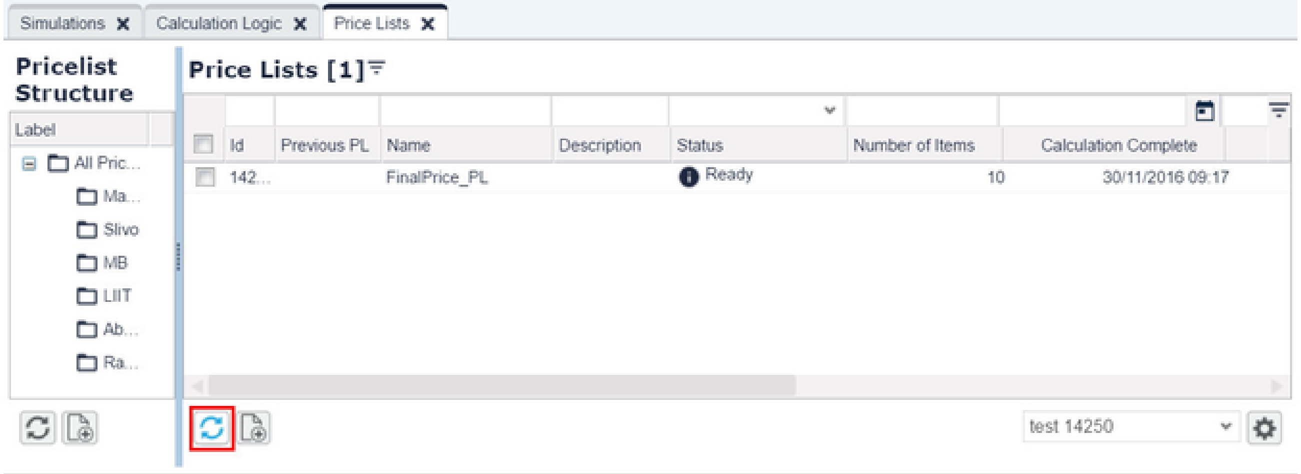
Matrix logic : — This logic defines the secondary key set

Matrix logic element :

Output Elements [1]

☑	Element Name	Label	Displayed in Quote Configurator	Hide in item list
☑	FinalPrice	FinalPrice	☐	☐

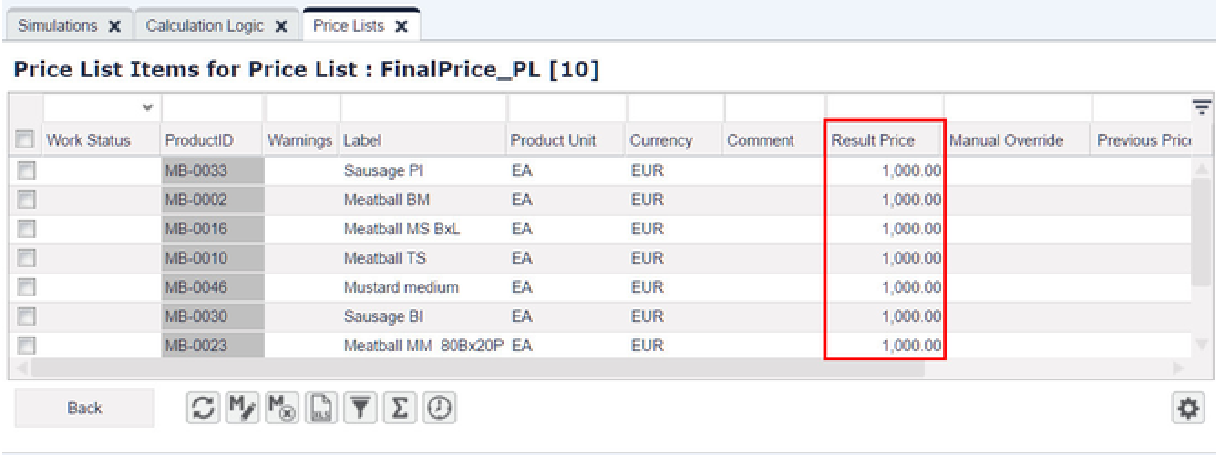
6. Click step 3 Create Price List to generate the Price List.
   Click the Refresh icon until your Price List has Status set to Ready.



Price Lists [1]

☐	Id	Previous PL	Name	Description	Status	Number of Items	Calculation Complete
☐	142...		FinalPrice_PL		Ready	10	30/11/2016 09:17

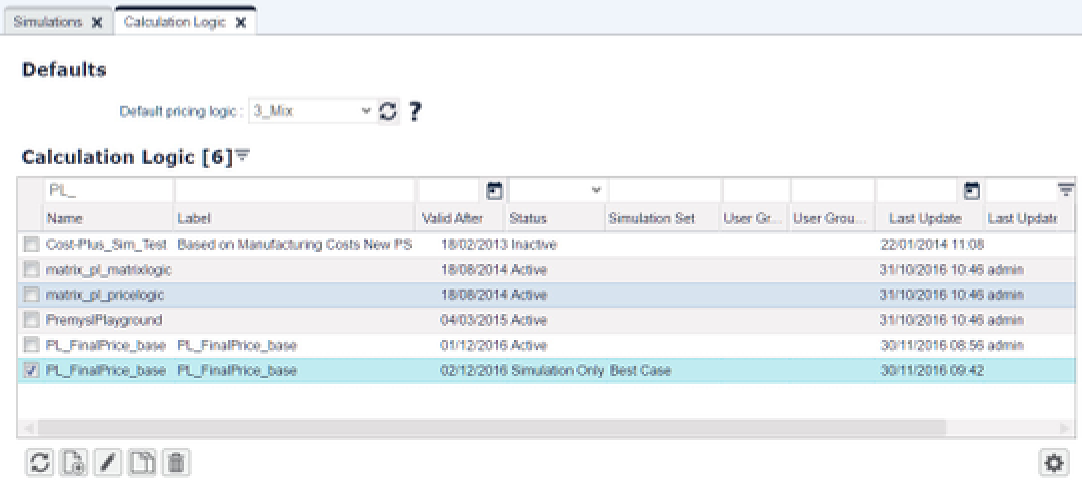
7. Check the Result Price:
   a. Select your Price List and click the Edit icon.
      You have a price list generated with prices generated for your products using your defined pricing logic.



Price List Items for Price List : FinalPrice_PL [10]

☐	Work Status	ProductID	Warnings	Label	Product Unit	Currency	Comment	Result Price	Manual Override	Previous Pric
☐		MB-0033		Sausage PI	EA	EUR		1,000.00		
☐		MB-0002		Meatball BM	EA	EUR		1,000.00		
☐		MB-0016		Meatball MS BxL	EA	EUR		1,000.00		
☐		MB-0010		Meatball TS	EA	EUR		1,000.00		
☐		MB-0046		Mustard medium	EA	EUR		1,000.00		
☐		MB-0030		Sausage BI	EA	EUR		1,000.00		
☐		MB-0023		Meatball MM 80Bx20P	EA	EUR		1,000.00		

8. Submit your price list and make sure it is in the Approved status.
9. Create two more pricing logics:
   a. Set their status as Simulation Only.
   b. Set the Valid After date for later than the actually valid Calculation Logic.



Simulations ✕ | Calculation Logic ✕

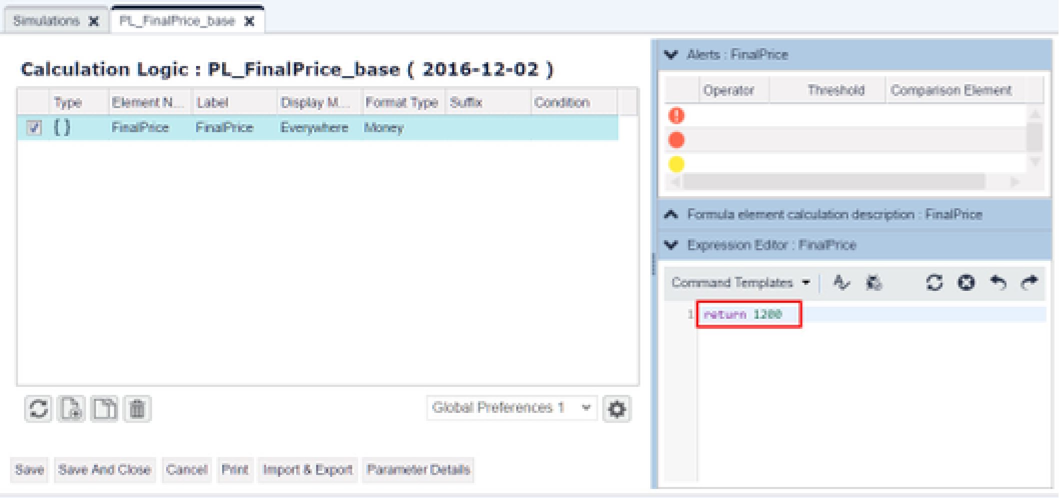
Defaults

Default pricing logic : 3_Mix

Calculation Logic [6]

	Name	Label	Valid After	Status	Simulation Set	User Gr...	User Grou...	Last Update	Last Updat
	PL_								
□	Cost-Plus_Sim_Test	Based on Manufacturing Costs New PS	18/02/2013	Inactive				22/01/2014 11:08	
□	matrix_pl_matrixlogic		18/08/2014	Active				31/10/2016 10:46	admin
□	matrix_pl_pricelogic		18/08/2014	Active				31/10/2016 10:46	admin
□	PremyslPlayground		04/03/2015	Active				31/10/2016 10:46	admin
□	PL_FinalPrice_base	PL_FinalPrice_base	01/12/2016	Active				30/11/2016 08:56	admin
☑	PL_FinalPrice_base	PL_FinalPrice_base	02/12/2016	Simulation Only	Best Case			30/11/2016 09:42	

c. Edit the pricing logic.
d. For each logic change the return value, so that you can easily distinguish which pricing logic was used for generation of the price list.



Simulations ✕ | PL_FinalPrice_base ✕

Calculation Logic : PL_FinalPrice_base (2016-12-02)

	Type	Element N...	Label	Display M...	Format Type	Suffix	Condition
☑	{ }	FinalPrice	FinalPrice	Everywhere	Money		

Alerts : FinalPrice

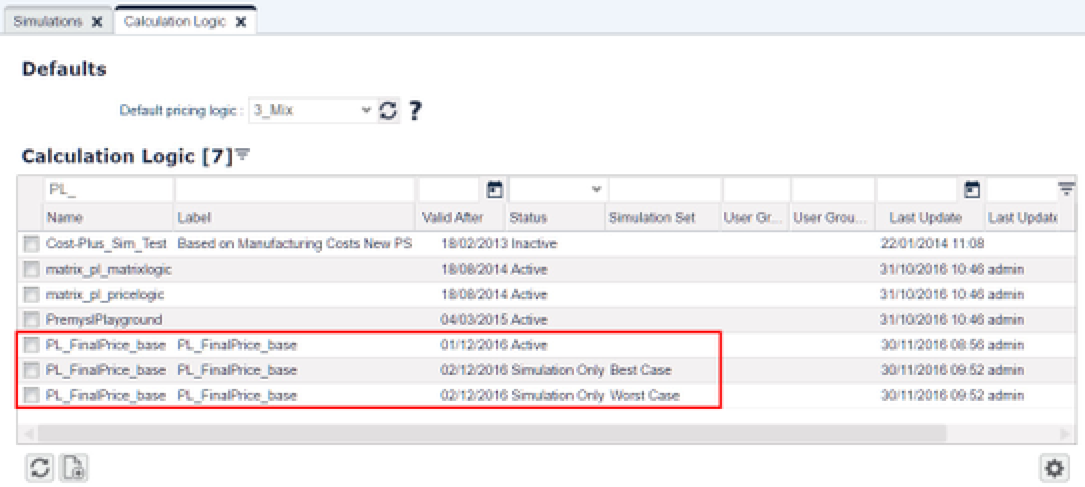
Operator	Threshold	Comparison Element

Formula element calculation description : FinalPrice

Expression Editor : FinalPrice

```
return 1200
```

e. The final setup should look like this:



Simulations ✕ | Calculation Logic ✕

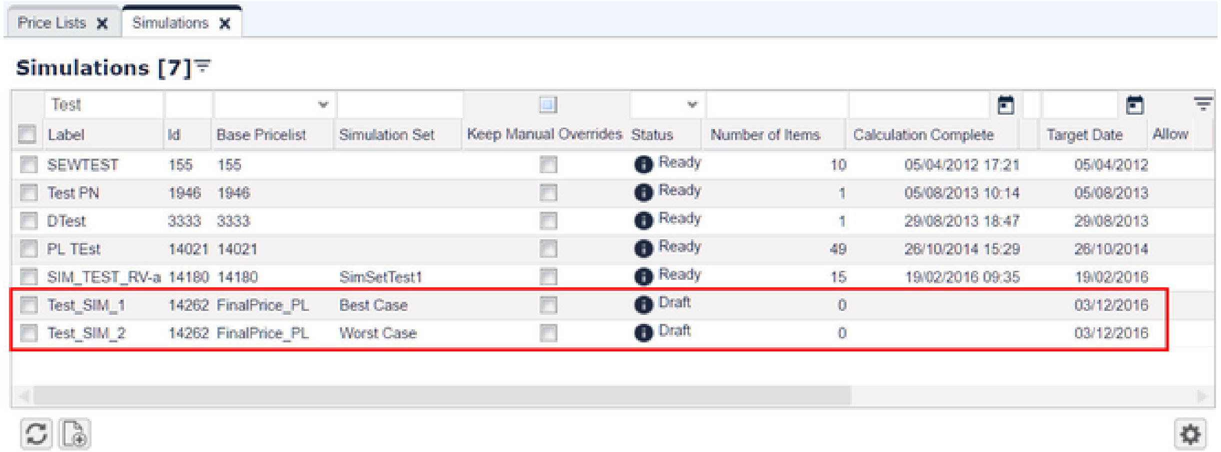
Defaults

Default pricing logic : 3_Mix

Calculation Logic [7]

	Name	Label	Valid After	Status	Simulation Set	User Gr...	User Grou...	Last Update	Last Updat
	PL_								
□	Cost-Plus_Sim_Test	Based on Manufacturing Costs New PS	18/02/2013	Inactive				22/01/2014 11:08	
□	matrix_pl_matrixlogic		18/08/2014	Active				31/10/2016 10:46	admin
□	matrix_pl_pricelogic		18/08/2014	Active				31/10/2016 10:46	admin
□	PremyslPlayground		04/03/2015	Active				31/10/2016 10:46	admin
□	PL_FinalPrice_base	PL_FinalPrice_base	01/12/2016	Active				30/11/2016 08:56	admin
□	PL_FinalPrice_base	PL_FinalPrice_base	02/12/2016	Simulation Only	Best Case			30/11/2016 09:52	admin
□	PL_FinalPrice_base	PL_FinalPrice_base	02/12/2016	Simulation Only	Worst Case			30/11/2016 09:52	admin

10. Go to PriceBuilder > Simulations.
    a. Create two new Simulations.
    b. In the Base Price List column, associate your Price List with the simulations.
    c. In the Simulation Set column, type the labels you have already used for your simulation price list.
    d. Set Target Date.



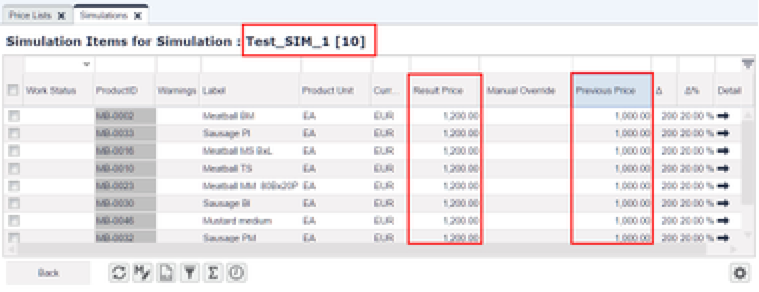
Simulations [7]

□	Label	Id	Base Pricelist	Simulation Set	Keep Manual Overrides	Status	Number of Items	Calculation Complete	Target Date	Allow
	Test									
□	SEWTEST	155	155		□	Ready	10	05/04/2012 17:21	05/04/2012	
□	Test PN	1946	1946		□	Ready	1	05/08/2013 10:14	05/08/2013	
□	DTest	3333	3333		□	Ready	1	29/08/2013 18:47	29/08/2013	
□	PL TEst	14021	14021		□	Ready	49	26/10/2014 15:29	26/10/2014	
□	SIM_TEST_RV-a	14180	14180	SimSetTest1	□	Ready	15	19/02/2016 09:35	19/02/2016	
□	Test_SIM_1	14262	FinalPrice_PL	Best Case	□	Draft	0		03/12/2016	
□	Test_SIM_2	14262	FinalPrice_PL	Worst Case	□	Draft	0		03/12/2016	

Be careful about the 'Valid After' date of both your active pricing logic and simulation pricing logic! The priorities are:

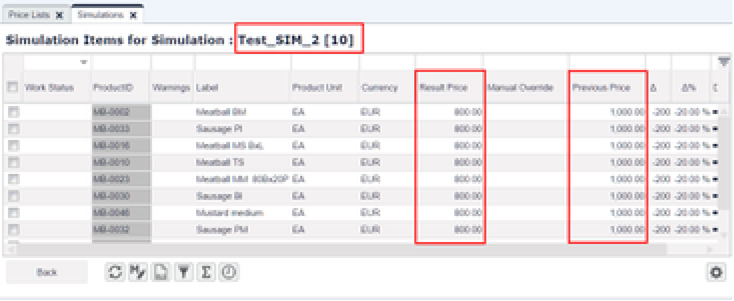
1. Latest Valid After date (i.e. if the date for simulation logic is sooner, it will not be taken into account when generating the simulation price list, it would take the active one).
2. If the Valid After dates are the same for active and simulation logics, then again the active one is used.
3. If there is Simulation Set name set for the simulation pricing logic, then this one is used.

11. Generate you Price List with associated simulation logic:
    a. Select your simulation and click the Calculate icon.
    b. Keep refreshing until the status is Ready.
12. Check the Result Price of the new (simulation) pricing logic:
    a. Click the View/Edit icon to view the result price after applying a new (simulation) logic.



Simulation Items for Simulation : Test_SIM_1 [10]

□	Work Status	ProductID	Warnings	Label	Product Unit	Curr...	Result Price	Manual Override	Previous Price	Δ	Δ%	Detail
□		MB-0002		Meatball BM	EA	EUR	1,200.00		1,000.00	200	20.00 %	➡
□		MB-0033		Sausage PI	EA	EUR	1,200.00		1,000.00	200	20.00 %	➡
□		MB-0016		Meatball MS BxL	EA	EUR	1,200.00		1,000.00	200	20.00 %	➡
□		MB-0010		Meatball TS	EA	EUR	1,200.00		1,000.00	200	20.00 %	➡
□		MB-0023		Meatball MM 80Bx20P	EA	EUR	1,200.00		1,000.00	200	20.00 %	➡
□		MB-0030		Sausage BI	EA	EUR	1,200.00		1,000.00	200	20.00 %	➡
□		MB-0046		Mustard medium	EA	EUR	1,200.00		1,000.00	200	20.00 %	➡
□		MB-0032		Sausage PM	EA	EUR	1,200.00		1,000.00	200	20.00 %	➡



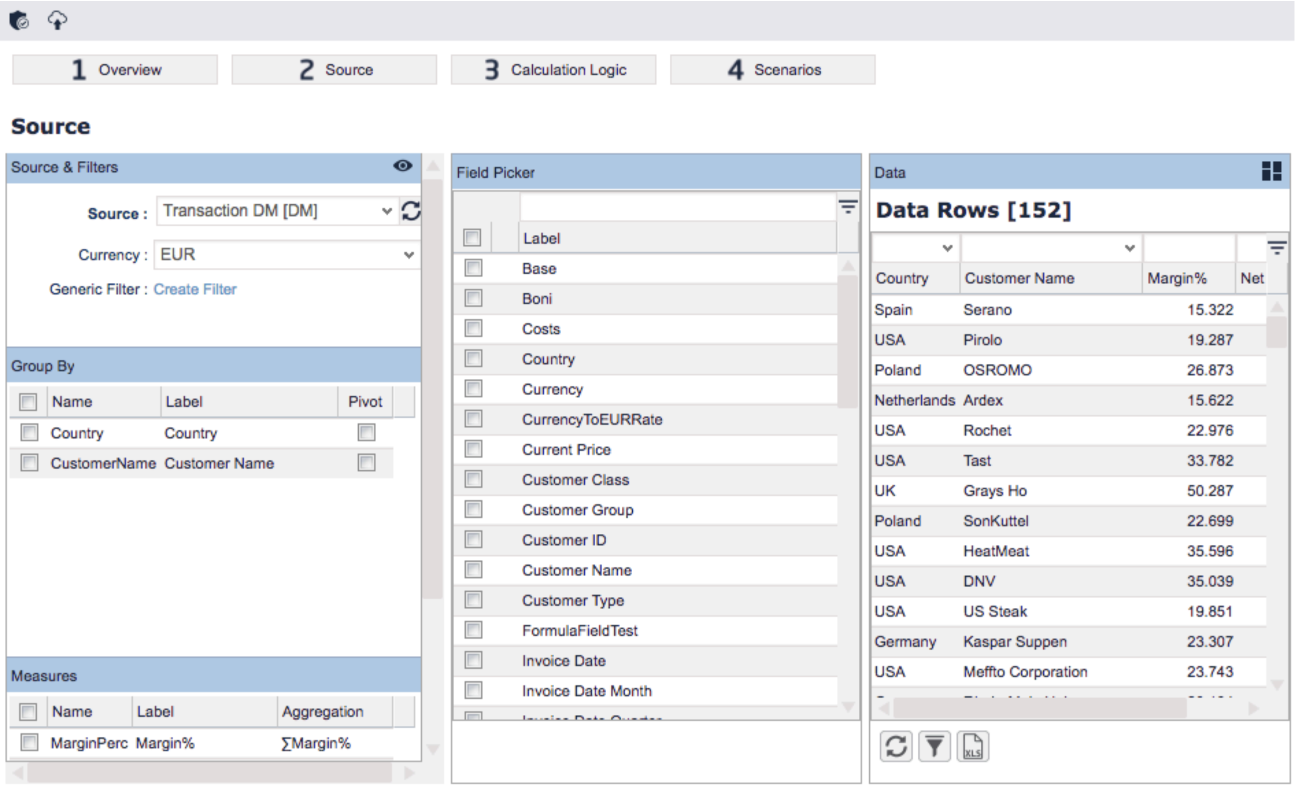
Simulation Items for Simulation : Test_SIM_2 [10]

□	Work Status	ProductID	Warnings	Label	Product Unit	Currency	Result Price	Manual Override	Previous Price	Δ	Δ%
□		MB-0002		Meatball BM	EA	EUR	800.00		1,000.00	-200	-20.00 %
□		MB-0033		Sausage PI	EA	EUR	800.00		1,000.00	-200	-20.00 %
□		MB-0016		Meatball MS BxL	EA	EUR	800.00		1,000.00	-200	-20.00 %
□		MB-0010		Meatball TS	EA	EUR	800.00		1,000.00	-200	-20.00 %
□		MB-0023		Meatball MM 80Bx20P	EA	EUR	800.00		1,000.00	-200	-20.00 %
□		MB-0030		Sausage BI	EA	EUR	800.00		1,000.00	-200	-20.00 %
□		MB-0046		Mustard medium	EA	EUR	800.00		1,000.00	-200	-20.00 %
□		MB-0032		Sausage PM	EA	EUR	800.00		1,000.00	-200	-20.00 %

How to Run a Simulation in PriceAnalyzer

This section explains how to create simulations in PriceAnalyzer and how to read and compare the results for different scenarios.

1. Create a new simulation by clicking the Add icon. Give it a name and a label.
2. Go to the Source section and in the Source field, select the Datamart whose data you want to use for the simulation.
3. Still in the Source section, create a data query to get the right data set for the simulation. The available options are the same as in Rollups. When you are done and before you continue to the next step, validate and deploy the simulation.



Source

Source & Filters

Source : Transaction DM [DM]

Currency : EUR

Generic Filter : Create Filter

Group By

□	Name	Label	Pivot
□	Country	Country	□
□	CustomerName	Customer Name	□

Measures

□	Name	Label	Aggregation
□	MarginPerc	Margin%	∑Margin%

Field Picker

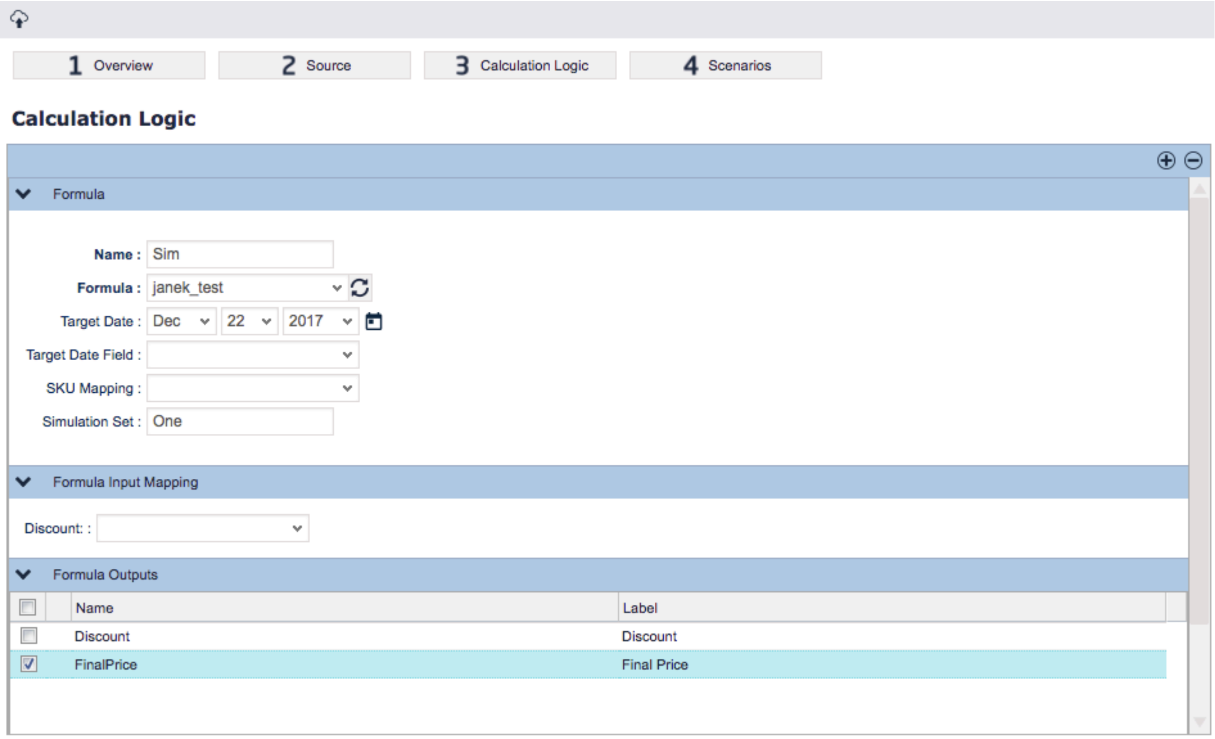
□	Label
□	Base
□	Boni
□	Costs
□	Country
□	Currency
□	CurrencyToEURRate
□	Current Price
□	Customer Class
□	Customer Group
□	Customer ID
□	Customer Name
□	Customer Type
□	FormulaFieldTest
□	Invoice Date
□	Invoice Date Month

Data

Data Rows [152]

Country	Customer Name	Margin%	Net
Spain	Serano	15.322	
USA	Pirolo	19.287	
Poland	OSROMO	26.873	
Netherlands	Ardex	15.622	
USA	Rochet	22.976	
USA	Tast	33.782	
UK	Grays Ho	50.287	
Poland	SonKuttel	22.699	
USA	HeatMeat	35.596	
USA	DNV	35.039	
USA	US Steak	19.851	
Germany	Kaspar Suppen	23.307	
USA	Meffto Corporation	23.743	

4. Continue to the Calculation Logic section and:
   a. Select the formula that will manipulate the data.
   b. Set the target date either manually or select a Datamart field containing the target dates.
   c. (Optional) Map SKU to a Datamart field.
   d. (Optional) Enter the name of a Simulation Set – a label that determines which pricing parameters will be used by the logic (those with the same simulation set label).
   e. In Formula Input Mapping, you can map user inputs from the selected calculation logic to the corresponding Datamart fields. You can choose to make no selection – any formula input not mapped here can be set later on the Scenario level.
      The idea behind is that in this section you define the inputs that are static for all scenarios and on the Scenario level you set variables by which you compare the scenarios. For example, you can increase 'Cost' by 10 points for each next scenario.
   f. Review the Formula Outputs panel, which lists the calculation logic outputs as defined by its elements. Selected outputs will be a dded to the result data.
      Deploy the simulation before you continue.



Calculation Logic

Formula

Name : Sim
Formula : janek_test
Target Date : Dec 22 2017
Target Date Field :
SKU Mapping :
Simulation Set : One

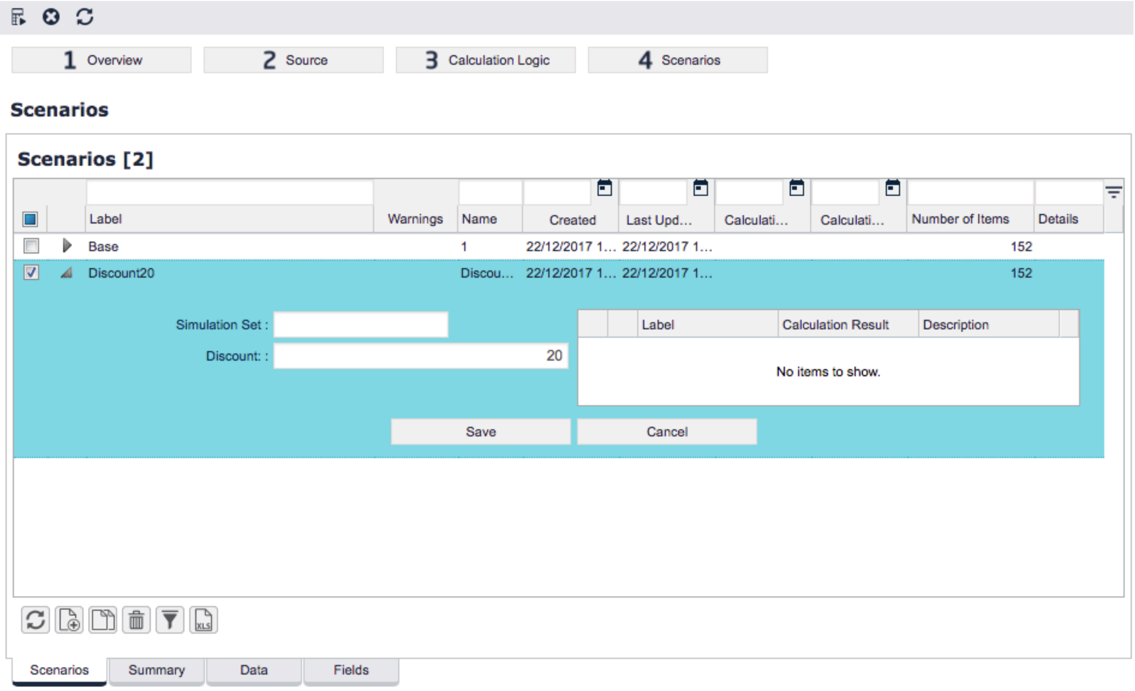
Formula Input Mapping

Discount: :

Formula Outputs

☐	Name	Label
☐	Discount	Discount
☑	FinalPrice	Final Price

5. Go to the Scenarios section and define your scenarios. Each scenario should represent a different version of the simulation.
   a. (Optional) Enter the name of a Simulation Set, determining which pricing parameters will be used for the scenario.
   b. Enter the values for all the user inputs that have not been previously mapped. Typically, you will set a different value for each scenario.
   c. Run the simulation by clicking the Calculate icon.
   d. Review the results. For each scenario, there is a table with calculation results.
   There are also three additional tabs in this section:
      - **Summary** – When a simulation is completed, the resulting data can be summarized on various levels in the Summary tab. Here we essentially define Rollups to aggregate the data up to the desired level.
        The 'Base' rollup is created automatically by the application, both as a typical 'grand totals' sort of query (calculates the SUM() for each numeric field in the data set) and also as a template for a user-defined rollup.
        If averages rather than totals, or any other aggregation function are required, then the 'Base' rollup can be modified by clicking the Toggle button. It is, however, recommended saving the modified query as a new rollup. Similarly, when for example additional group-by's are needed, it is best to pick an existing rollup, add the group-by fields, and save it under a different name (see Rollups for details).
      - **Data** – Displays a data table containing the combination of the source rows and row level calculation results. In other words, each row resulting from the source query is enriched with the simulation's selected outputs' calculation. Note that if multiple scenarios are defined, there will be rows for all these scenarios. To inspect just one, you can filter the data set in the usual way.
      - **Fields** – Displays the source fields (from the query definition) and the selected outputs.



Scenarios

Scenarios [2]

☐	Label	Warnings	Name	Created	Last Upd…	Calculati…	Calculati…	Number of Items	Details
☐	Base		1	22/12/2017 1…	22/12/2017 1…			152	
☑	Discount20		Discou…	22/12/2017 1…	22/12/2017 1…			152	

Simulation Set :

Discount: : 20

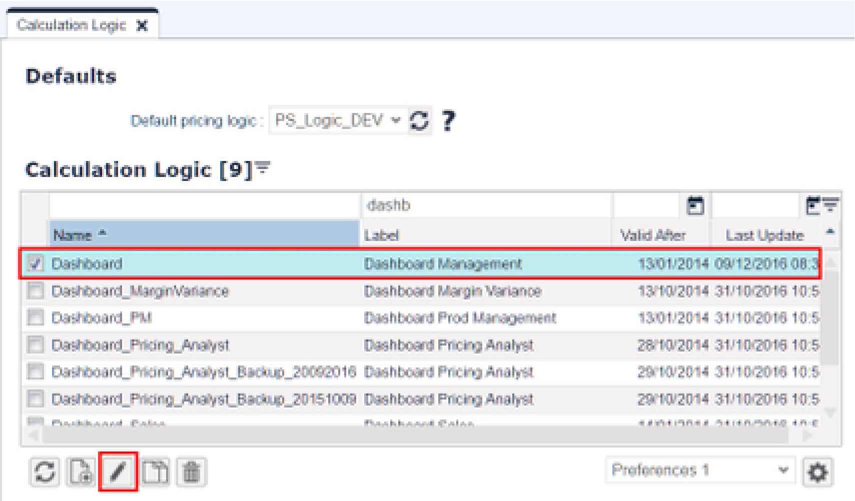
Label	Calculation Result	Description

No items to show.

How to Create New Dashboard

To create a new Dashboard:

1. In Configuration > Calculation Logic > Generic Logic, create a new logic.



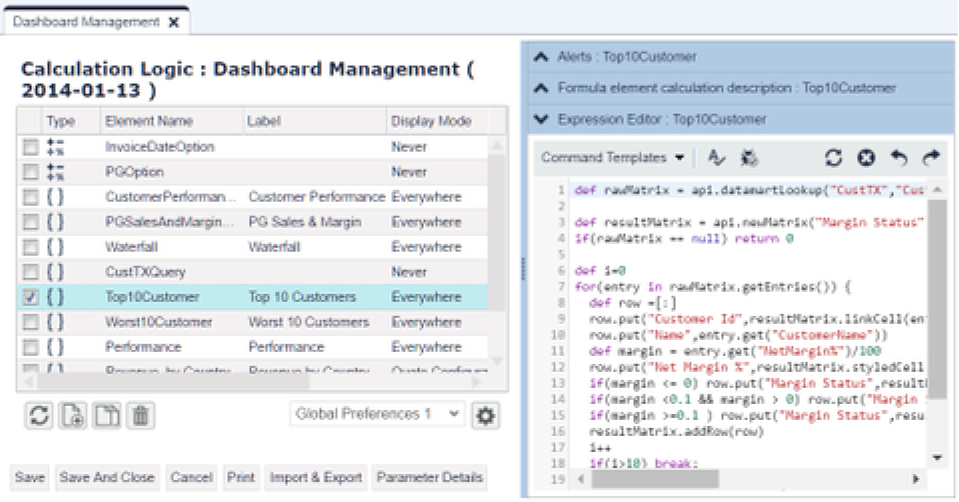
Defaults

Default pricing logic : PS_Logic_DEV

Calculation Logic [9]

	Name	Label	Valid After	Last Update
		dashb		
☑	Dashboard	Dashboard Management	13/01/2014	09/12/2016 08:3
☐	Dashboard_MarginVariance	Dashboard Margin Variance	13/10/2014	31/10/2016 10:5
☐	Dashboard_PM	Dashboard Prod Management	13/01/2014	31/10/2016 10:5
☐	Dashboard_Pricing_Analyst	Dashboard Pricing Analyst	28/10/2014	31/10/2016 10:5
☐	Dashboard_Pricing_Analyst_Backup_20092016	Dashboard Pricing Analyst	29/10/2014	31/10/2016 10:5
☐	Dashboard_Pricing_Analyst_Backup_20151009	Dashboard Pricing Analyst	29/10/2014	31/10/2016 10:5
☐	Dashboard_Sales	Dashboard Sales	14/01/2014	31/10/2016 10:5

2. Open the new logic and create elements which provide the content of portlets and ask for user input.

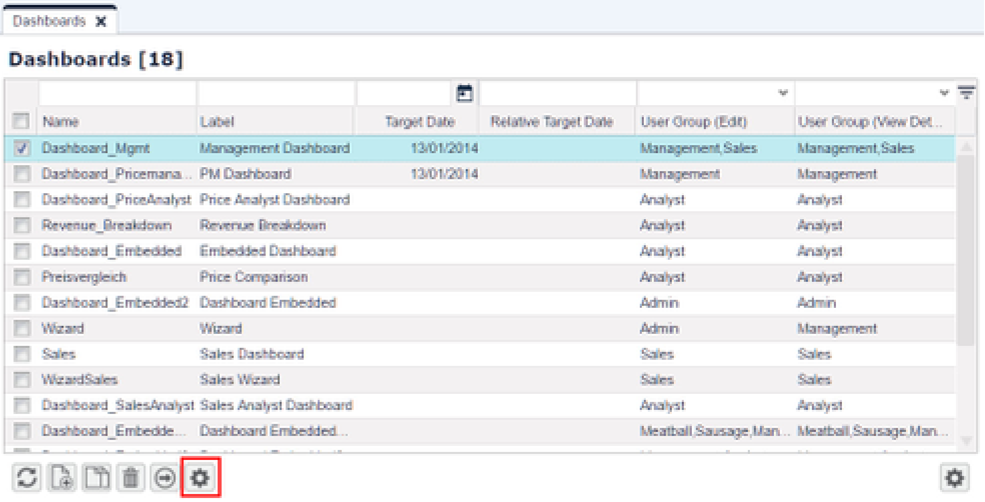


Calculation Logic : Dashboard Management (2014-01-13)

	Type	Element Name	Label	Display Mode
□	±%	InvoiceDateOption		Never
□	±%	PGOption		Never
□	{ }	CustomerPerforman...	Customer Performance	Everywhere
□	{ }	PGSalesAndMargin...	PG Sales & Margin	Everywhere
□	{ }	Waterfall	Waterfall	Everywhere
□	{ }	CustTXQuery		Never
☑	{ }	Top10Customer	Top 10 Customers	Everywhere
□	{ }	Worst10Customer	Worst 10 Customers	Everywhere
□	{ }	Performance	Performance	Everywhere
□	{ }	Revenue by Country	Revenue by Country	Quote Configur...

```
def rawMatrix = api.datamartLookup("CustTX","Cus
                                                
def resultMatrix = api.newMatrix("Margin Status"
if(rawMatrix == null) return 0

def i=0
for(entry in rawMatrix.getEntries()) {
  def row =[:]
  row.put("Customer Id",resultMatrix.linkCell(en
  row.put("Name",entry.get("CustomerName"))
  def margin = entry.get("NetMargin%")/100
  row.put("Net Margin %",resultMatrix.styledCell
  if(margin <= 0) row.put("Margin Status",result
  if(margin <0.1 && margin > 0) row.put("Margin 
  if(margin >=0.1 ) row.put("Margin Status",resu
  resultMatrix.addRow(row)
  i++
  if(i>10) break;
```

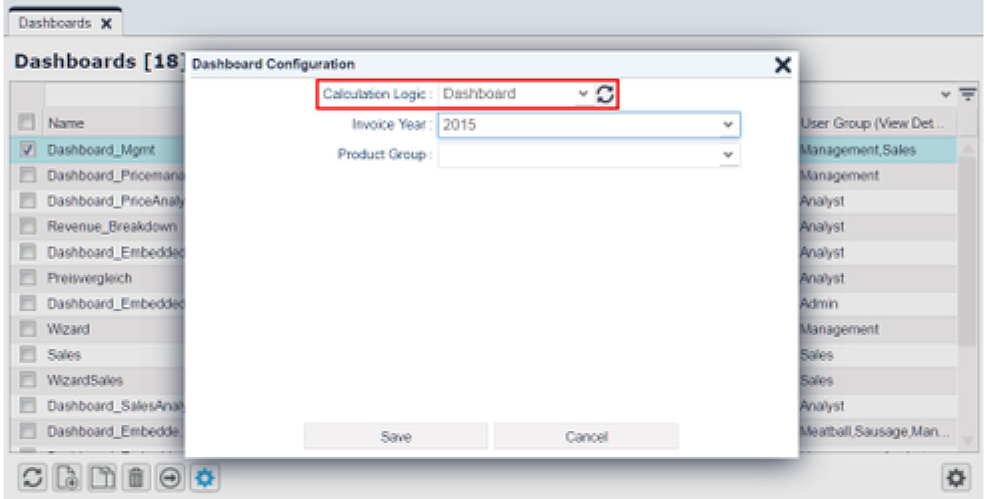
3. In Configuration > Dashboards create a new dashboard and click the Configure icon.



Dashboards [18]

	Name	Label	Target Date	Relative Target Date	User Group (Edit)	User Group (View Det...
☑	Dashboard_Mgmt	Management Dashboard	13/01/2014		Management,Sales	Management,Sales
□	Dashboard_Pricemana...	PM Dashboard	13/01/2014		Management	Management
□	Dashboard_PriceAnalyst	Price Analyst Dashboard			Analyst	Analyst
□	Revenue_Breakdown	Revenue Breakdown			Analyst	Analyst
□	Dashboard_Embedded	Embedded Dashboard			Analyst	Analyst
□	Preisvergleich	Price Comparison			Analyst	Analyst
□	Dashboard_Embedded2	Dashboard Embedded			Admin	Admin
□	Wizard	Wizard			Admin	Management
□	Sales	Sales Dashboard			Sales	Sales
□	WizardSales	Sales Wizard			Sales	Sales
□	Dashboard_SalesAnalyst	Sales Analyst Dashboard			Analyst	Analyst
□	Dashboard_Embedde...	Dashboard Embedded...			Meatball,Sausage,Man...	Meatball,Sausage,Man...

4. Assign your logic to the dashboard.



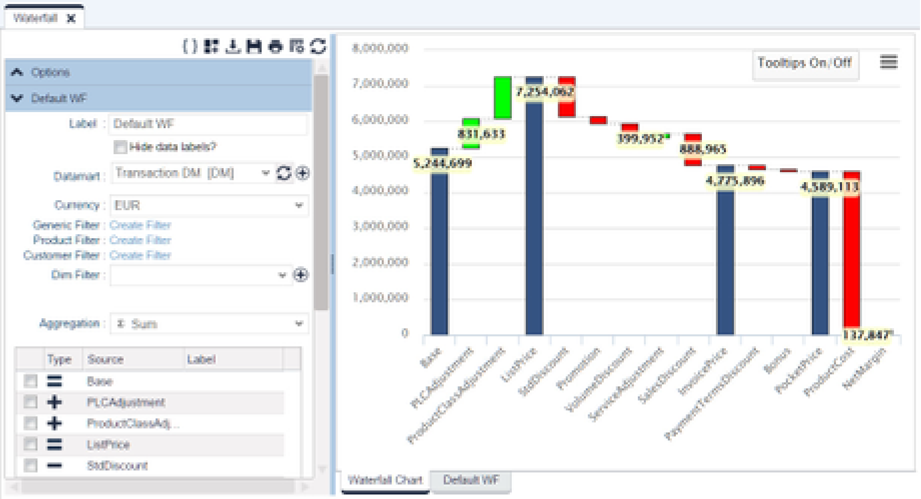
Dashboard Configuration

Calculation Logic : Dashboard

Invoice Year : 2015

Product Group :

5. In PriceAnalyzer > Data Analyzer and create a new Waterfall chart.



Label : Default WF

□ Hide data labels?

Datamart : Transaction DM [DM]

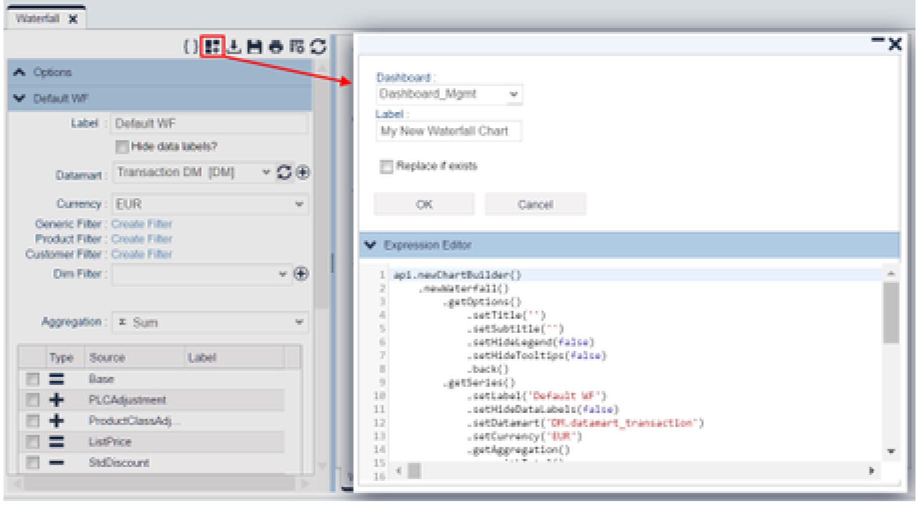
Currency : EUR

Generic Filter : Create Filter
Product Filter : Create Filter
Customer Filter : Create Filter
Dim Filter :

Aggregation : Σ Sum

	Type	Source	Label
□	=	Base	
□	+	PLCAdjustment	
□	+	ProductClassAdj...	
□	=	ListPrice	
□	−	StdDiscount	

6. Add the chart into the Dashboard (into the list of Portlets).



Label : Default WF

□ Hide data labels?

Datamart : Transaction DM [DM]

Currency : EUR

Generic Filter : Create Filter
Product Filter : Create Filter
Customer Filter : Create Filter
Dim Filter :

Aggregation : Σ Sum

	Type	Source	Label
□	=	Base	
□	+	PLCAdjustment	
□	+	ProductClassAdj...	
□	=	ListPrice	
□	−	StdDiscount	

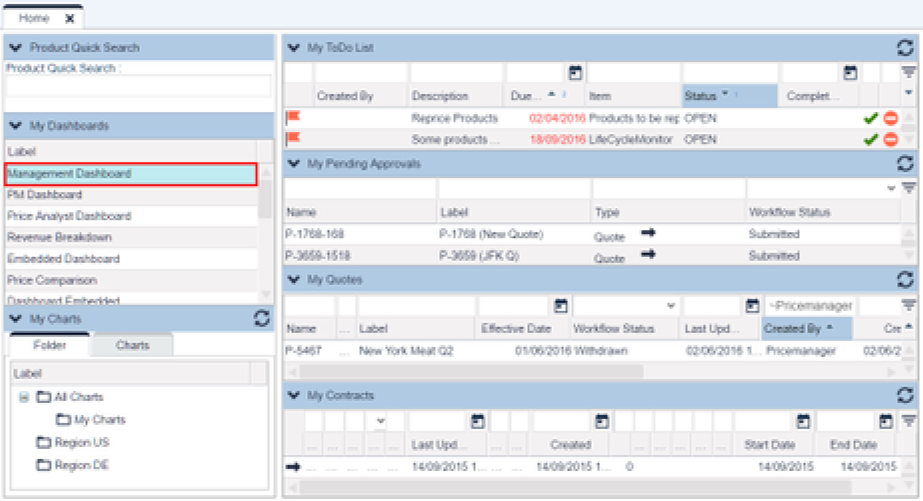
Dashboard : Dashboard_Mgmt

Label : My New Waterfall Chart

□ Replace if exists

```
api.newChartBuilder()
    .newWaterfall()
        .getOptions()
            .setTitle('')
            .setSubtitle('')
            .setHideLegend(false)
            .setHideTooltips(false)
            .back()
        .getSeries()
            .setLabel('Default WF')
            .setHideDataLabels(false)
            .setDatamart('DM.datamart_transaction')
            .setCurrency('EUR')
            .getAggregation()
```

7. Open the Dashboard from the Home page.



Label
Management Dashboard
PM Dashboard
Price Analyst Dashboard
Revenue Breakdown
Embedded Dashboard
Price Comparison
Dashboard Embedded

Label
All Charts
My Charts
Region US
Region DE

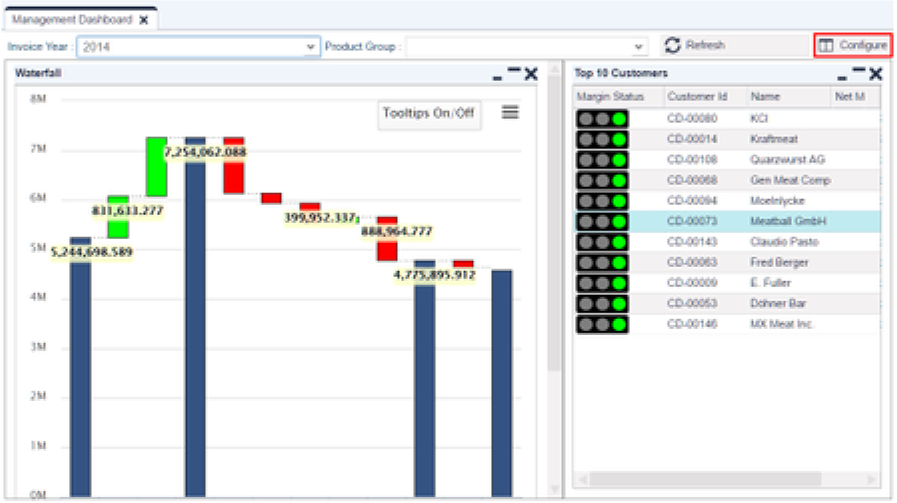
	Created By	Description	Due...	Item	Status	Complet...
		Reprice Products	02/04/2016	Products to be rep	OPEN	
		Some products ...	18/09/2016	LifeCycleMonitor	OPEN	

Name	Label	Type	Workflow Status
P-1768-168	P-1768 (New Quote)	Quote	Submitted
P-3659-1518	P-3659 (JFK Q)	Quote	Submitted

Name	...	Label	Effective Date	Workflow Status	Last Upd...	Created By	Cre
						~Pricemanager	
P-5467	...	New York Meat Q2	01/06/2016	Withdrawn	02/06/2016 1...	Pricemanager	02/06/2

...	...	...	...	...	Last Upd...	...	...	Created	...	...	...	...	...	Start Date	End Date
...	...	...	...	...	14/09/2015 1...	...	...	14/09/2015 1...	0					14/09/2015	14/09/2015

8. Set up the layout and place the Portlets (using the Configure button).



Invoice Year : 2014
Product Group :

Top 10 Customers

Margin Status	Customer Id	Name	Net M
●●●	CD-00080	KCI	
●●●	CD-00014	Kraftmeat	
●●●	CD-00108	Quarzwurst AG	
●●●	CD-00068	Gen Meat Comp	
●●●	CD-00094	Moelnlycke	
●●●	CD-00073	Meatball GmbH	
●●●	CD-00143	Claudio Pasto	
●●●	CD-00063	Fred Berger	
●●●	CD-00009	E. Fuller	
●●●	CD-00053	Dohner Bar	
●●●	CD-00146	MX Meat Inc.	

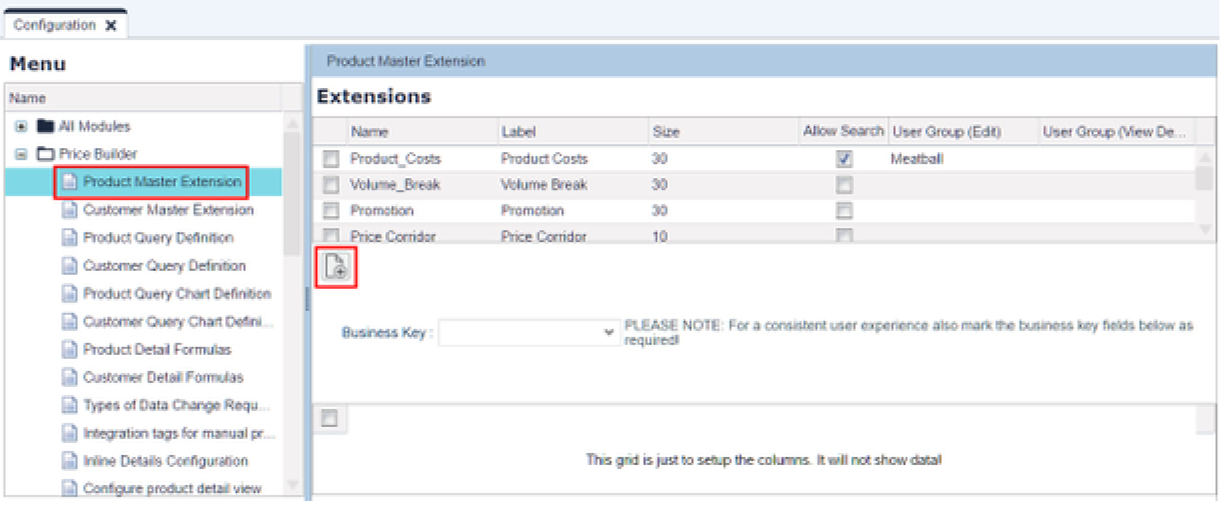
9. In the Configure menu choose Save Preferences and save the dashboard.

See also more advanced dashboard topics:

- How to Create Revenue Waterfall Dashboard
- How to Dynamically Generate Server Request from Dashboard
- How to Create an Interactive Embedded Dashboard
- How to Add a Quote Link to Dashboard

How to Create a New Product Extension

1. Navigate to Configuration > Product Master Extension.
2. Click the Add icon to create a new Extension (table).


Configuration
Menu
Name
All Modules
Price Builder
Product Master Extension
Customer Master Extension
Product Query Definition
Customer Query Definition
Product Query Chart Definition
Customer Query Chart Defini...
Product Detail Formulas
Customer Detail Formulas
Types of Data Change Requ...
Integration tags for manual pr...
Inline Details Configuration
Configure product detail view
Product Master Extension
Extensions

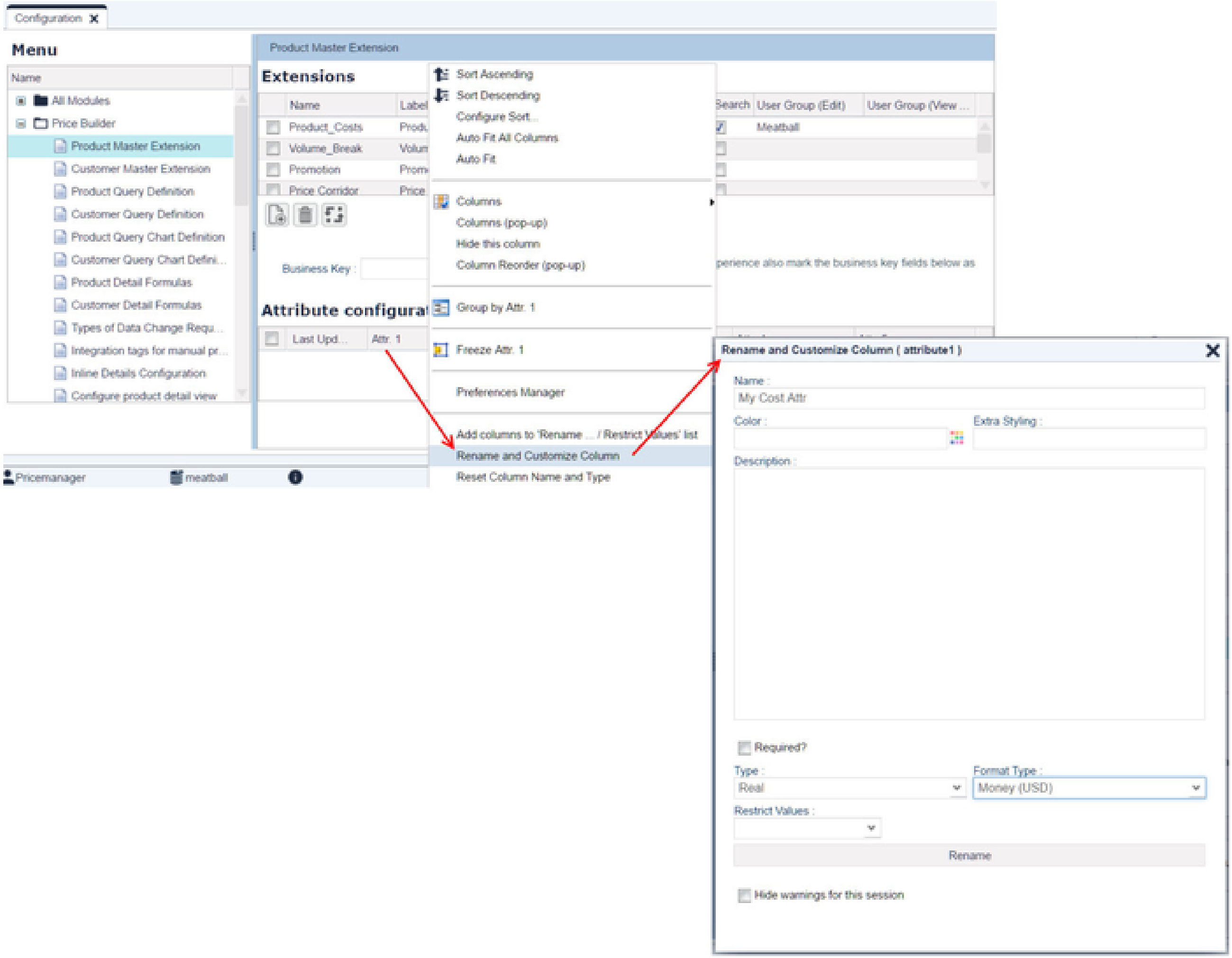
	Name	Label	Size	Allow Search	User Group (Edit)	User Group (View De...
□	Product_Costs	Product Costs	30	☑	Meatball	
□	Volume_Break	Volume Break	30	□		
□	Promotion	Promotion	30	□		
□	Price Corridor	Price Corridor	10	□		

Business Key :

PLEASE NOTE: For a consistent user experience also mark the business key fields below as required!

This grid is just to setup the columns. It will not show data!

3. Set up the columns you need.

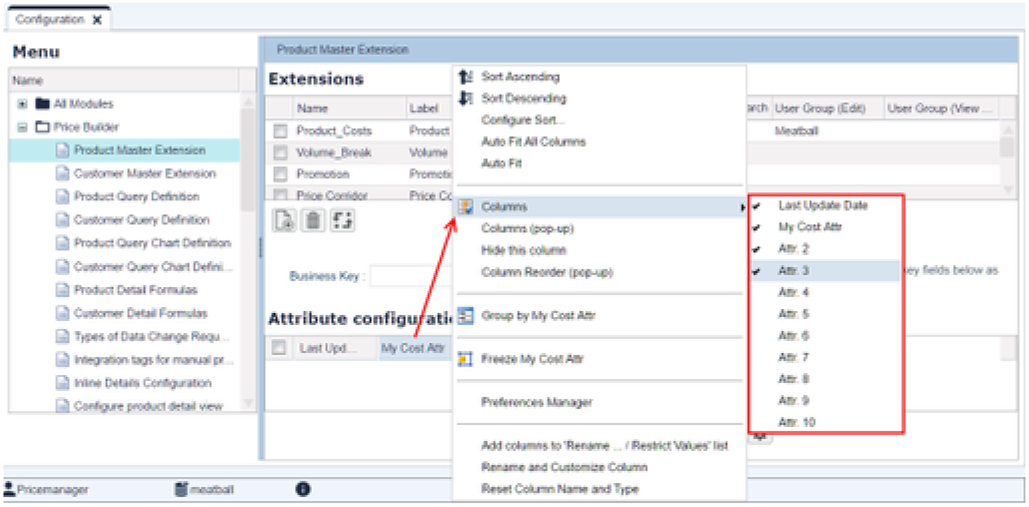


Name : My Cost Attr

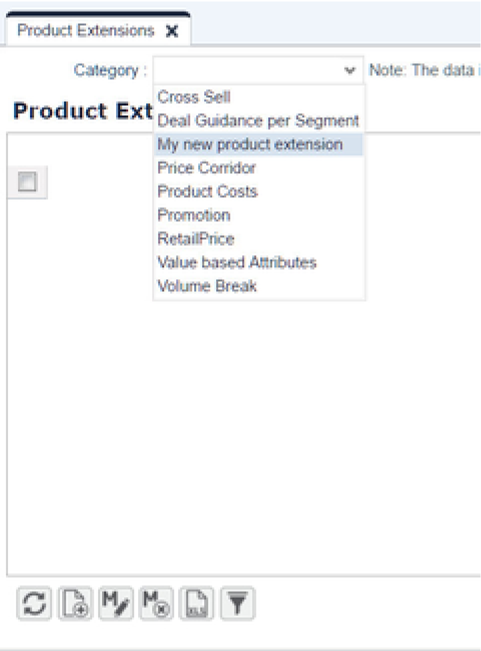
Type : Real

Format Type : Money (USD)

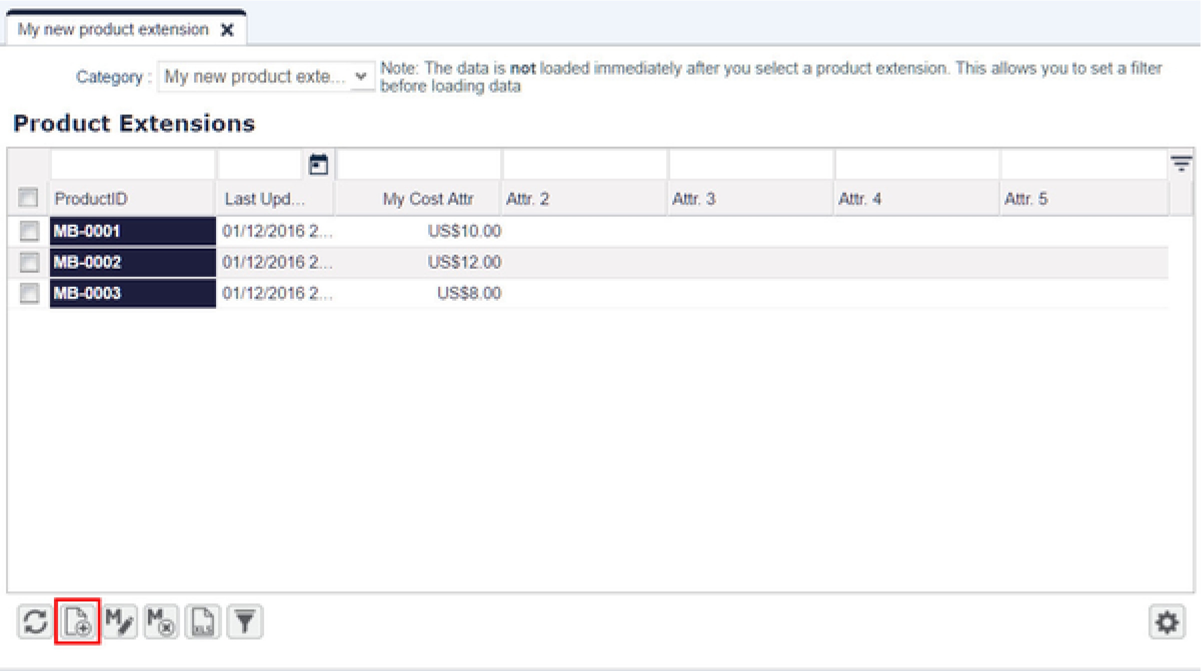
4. Hide all unused columns.



5. Log out and log in (to see your new Extension).
6. Navigate to Master Data > Product Extensions and select your new extension.



7. Add data to the Product Extension: either add a couple of lines for existing Products or import them from an Excel file.



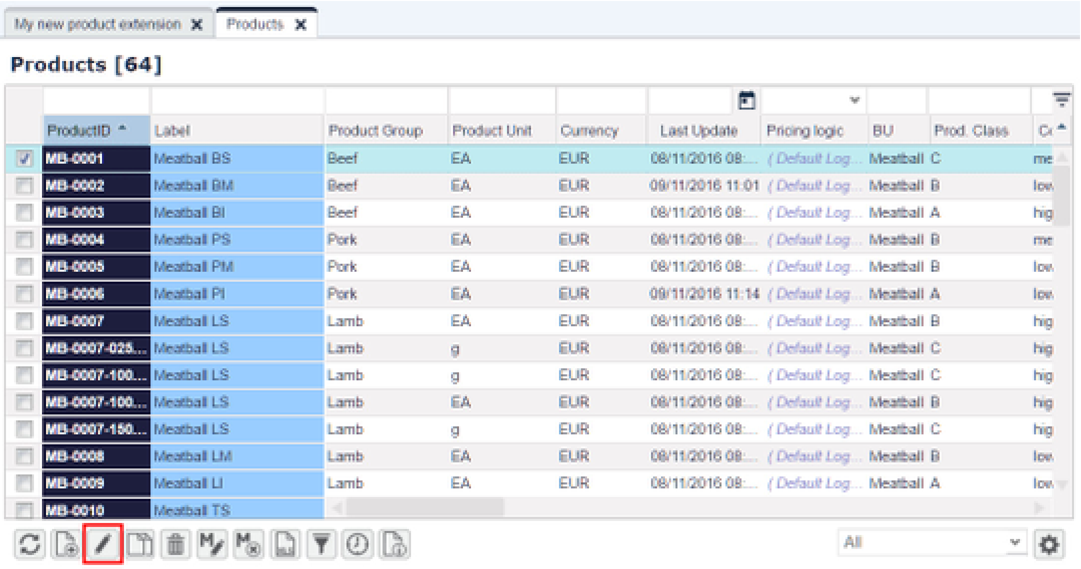
Category : My new product exte...

Note: The data is **not** loaded immediately after you select a product extension. This allows you to set a filter before loading data

Product Extensions

ProductID	Last Upd...	My Cost Attr	Attr. 2	Attr. 3	Attr. 4	Attr. 5
MB-0001	01/12/2016 2...	US$10.00				
MB-0002	01/12/2016 2...	US$12.00				
MB-0003	01/12/2016 2...	US$8.00				

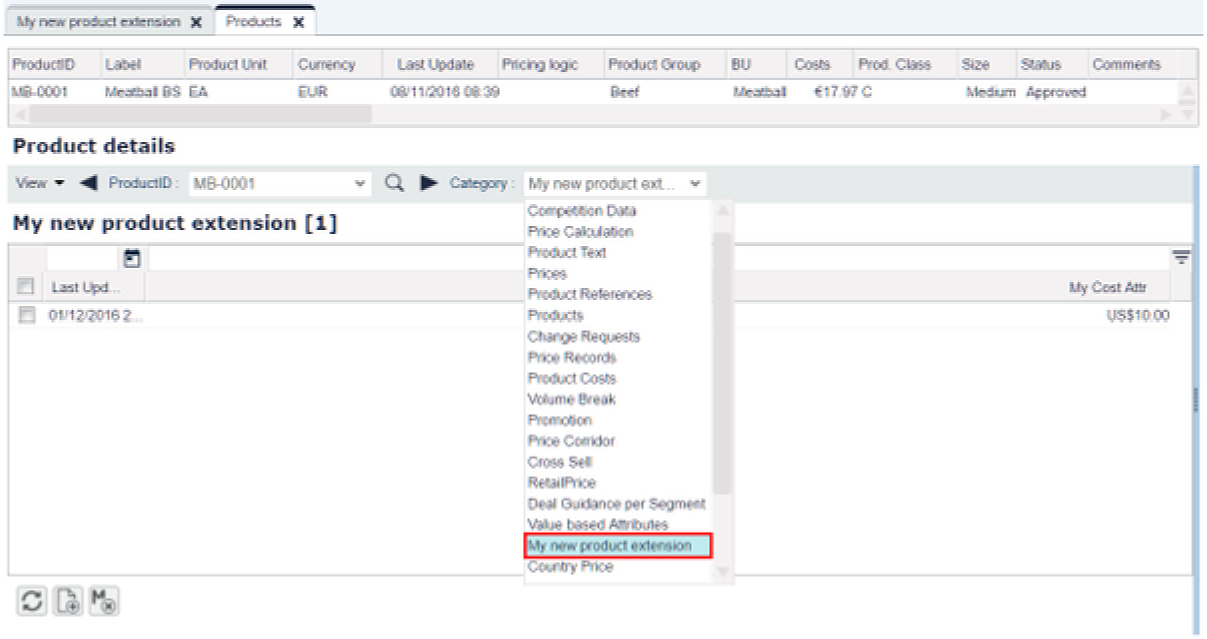
8. Add the Extension to the Product Detail page.
    a. Navigate to Master Data > Products.
    b. Select the product and click the Edit icon.



Products [64]

	ProductID	Label	Product Group	Product Unit	Currency	Last Update	Pricing logic	BU	Prod. Class	C
☑	MB-0001	Meatball BS	Beef	EA	EUR	08/11/2016 08:...	(Default Log...	Meatball	C	me
□	MB-0002	Meatball BM	Beef	EA	EUR	09/11/2016 11:01	(Default Log...	Meatball	B	low
□	MB-0003	Meatball BI	Beef	EA	EUR	08/11/2016 08:...	(Default Log...	Meatball	A	hig
□	MB-0004	Meatball PS	Pork	EA	EUR	08/11/2016 08:...	(Default Log...	Meatball	B	me
□	MB-0005	Meatball PM	Pork	EA	EUR	08/11/2016 08:...	(Default Log...	Meatball	B	low
□	MB-0006	Meatball PI	Pork	EA	EUR	09/11/2016 11:14	(Default Log...	Meatball	A	low
□	MB-0007	Meatball LS	Lamb	EA	EUR	08/11/2016 08:...	(Default Log...	Meatball	B	hig
□	MB-0007-025...	Meatball LS	Lamb	g	EUR	08/11/2016 08:...	(Default Log...	Meatball	C	hig
□	MB-0007-100...	Meatball LS	Lamb	g	EUR	08/11/2016 08:...	(Default Log...	Meatball	C	hig
□	MB-0007-100...	Meatball LS	Lamb	EA	EUR	08/11/2016 08:...	(Default Log...	Meatball	B	hig
□	MB-0007-150...	Meatball LS	Lamb	g	EUR	08/11/2016 08:...	(Default Log...	Meatball	C	hig
□	MB-0008	Meatball LM	Lamb	EA	EUR	08/11/2016 08:...	(Default Log...	Meatball	B	low
□	MB-0009	Meatball LI	Lamb	EA	EUR	08/11/2016 08:...	(Default Log...	Meatball	A	low
□	MB-0010	Meatball TS								

c. Select your new extension in the Category combo-box.



ProductID	Label	Product Unit	Currency	Last Update	Pricing logic	Product Group	BU	Costs	Prod. Class	Size	Status	Comments
MB-0001	Meatball BS	EA	EUR	08/11/2016 08:39		Beef	Meatball	€17.97	C	Medium	Approved	

Product details

View ▾ ◀ ProductID : MB-0001 ▶ Category : My new product ext...

My new product extension [1]

□	Last Upd...	My Cost Attr
□	01/12/2016 2...	US$10.00

9. Now you can see your extension in the Product Detail.

How to Create a Chart

You can display charts in dashboards, calculation results and other places in the application to graphically visualize selected performance indicators.

There are three ways to define a chart in Pricefx:

- PriceAnalyzer Charts – The basic and easy option, for which no programming skills are necessary. Simply define a chart in the UI and add it to a dashboard or copy and paste the definition to your calculation logic. On the downside, rendering can be slow in the Classic UI.
- FlexCharts – These are the legacy dashboard charts if PA Charts are not flexible enough. Templates can be created to shorten the Groovy code, and some level of JavaScript support allows for advanced number formatting. Rendering is fast, as the config goes straight to the Highcharts library. Server-side parsing helps catching errors, but unfortunately this solution relies on a 3rd party library that is based on the older 4.0 Highcharts release.
- Highcharts – Allow you to create highly flexible chart definitions with access to the full feature set of the latest Highcharts library release. Details of the chart are defined in a calculation logic, together with the data which can come either from PriceAnalyzer or other storages or can be created on the fly. The chart can then be used throughout the application.

How to Create a Highchart

A Highchart is defined in a calculation logic, using the `api.buildHighchart` method. Refer to the Highcharts API reference to explore the full variety of settings.

At the very least you need to provide `series[n].type` and `series[n].data`, or you can omit `series.type` if there is a default `chart.type` value.

If a message displays, saying that a JavaScript module may be missing, go to the Highcharts documentation and find out which modules are required for each chart type.

The following example fetches data from the Datamart and creates a variwide chart:

> If you cannot find an example of an existing logic that uses a chart type that you intend to use, it should be fairly easy to adapt one of the Highcharts Demos. Select your chart and click on the 'View Options' button and you will see the JS source code, including a `Highcharts.chart('container', {...})` call. This `{...}` content is the `definition` map. You will have to convert it from JS to Groovy. This always implies replacing the curly brackets by square brackets (`{} -> []`). But you may also sometimes encounter code like `Highcharts.color(colors[5]).brighten(0.2).get()` which you should replace by your own hard coded colors, or `formatter: function () {...}` which you would have to replace by a simpler pattern formatter `format: '{point.name}: {point.y:.1f}%'`.

> See also General Chart Settings for information about how to configure a global theme definition.

How to Upload CSV Data into PA

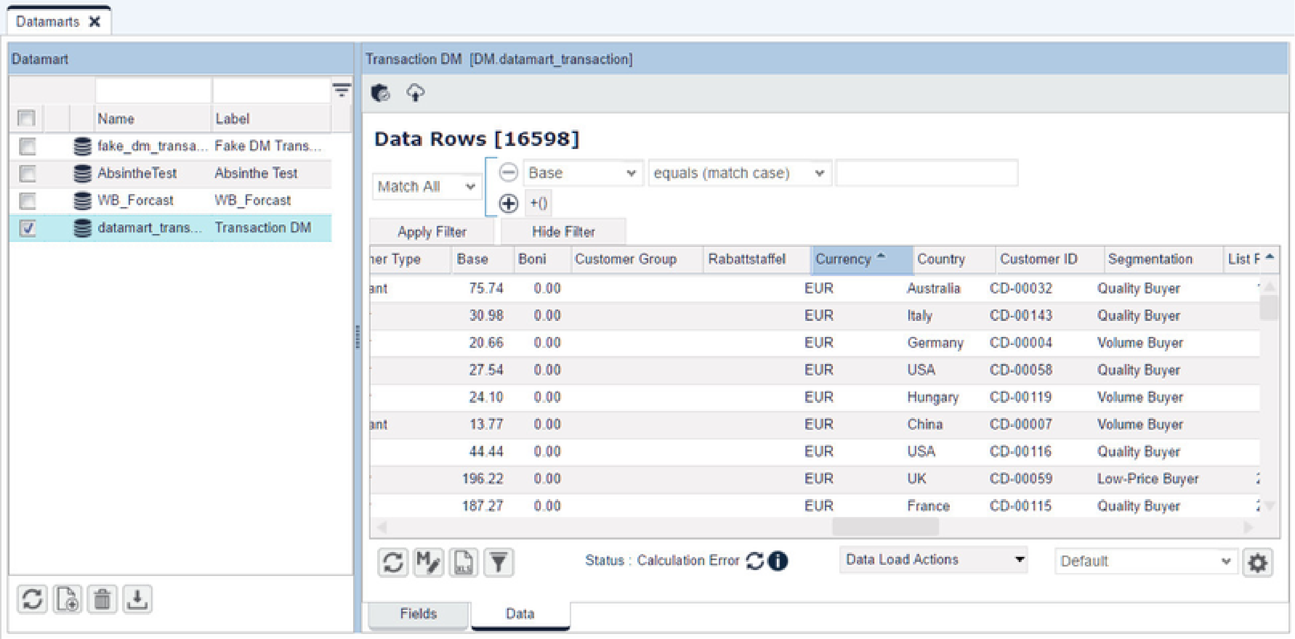
See the Pricefx documentation section Step-by-step: Import Data from Excel into PriceAnalyzer.

How to Set up Currencies

To prepare currencies to be used in logics, follow these steps:

1. Load currency data into the currency Data Source.
    - Dates in a currency act like a valid from date, so the rate for a given date holds until it finds a new rate for a new date. This prevents gaps in your exchange rates. The system derives the CcyValidTo date from the next row's CcyValidFrom date automatically. Therefore, you should not load CcyValidTo dates into the currency Data Source. CcyValidFrom and CcyValidTo are both inclusive, meaning that the exchange rate is applied to those dates, as well as any in between CcyValidFrom and CcyValidTo.
      Note: The underlying table actually contains rows for each individual day. (The exchange rates are looked up at query time.)
    - To ensure correct functioning during Datamart Refresh, values in the Money type fields must fulfill the following:
        a. Have currency conversions from the Base Currency (e.g. USD EUR).
        b. Have conversions back and forth (e.g. USD EUR, EUR USD).
        c. Have 1:1 conversions (e.g. USD USD).

2. Create a currency field in the target Data Source. This field marks which currency that row's Money fields are in. The currency field MUST have a 'Currency' data type. When a Data Source has a currency field, all Money fields are assumed to be denominated in the value of that currency field.
   ⚠ Only one Currency field is allowed per Data Source.



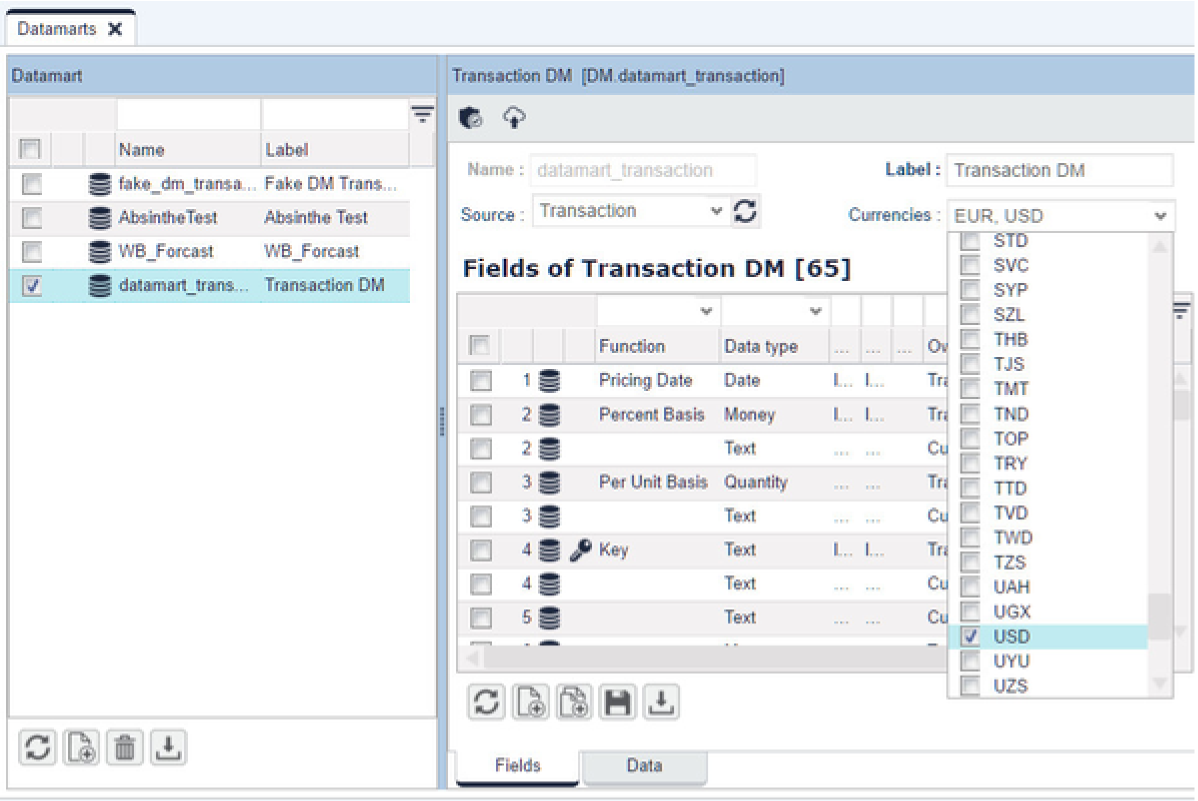
☐	Name	Label
☐	fake_dm_transa...	Fake DM Trans...
☐	AbsintheTest	Absinthe Test
☐	WB_Forcast	WB_Forcast
☑	datamart_trans...	Transaction DM

Data Rows [16598]

ner Type	Base	Boni	Customer Group	Rabattstaffel	Currency	Country	Customer ID	Segmentation	List F
ant	75.74	0.00			EUR	Australia	CD-00032	Quality Buyer	
	30.98	0.00			EUR	Italy	CD-00143	Quality Buyer	
	20.66	0.00			EUR	Germany	CD-00004	Volume Buyer	
	27.54	0.00			EUR	USA	CD-00058	Quality Buyer	
	24.10	0.00			EUR	Hungary	CD-00119	Volume Buyer	
ant	13.77	0.00			EUR	China	CD-00007	Volume Buyer	
	44.44	0.00			EUR	USA	CD-00116	Quality Buyer	
	196.22	0.00			EUR	UK	CD-00059	Low-Price Buyer	
	187.27	0.00			EUR	France	CD-00115	Quality Buyer	

3. Select all necessary currencies in the currency dropdown in the Fields tab of your Datamart and deploy the Datamart. The Datamart now

knows how to convert all those Money fields to your target 'reporting' currency.



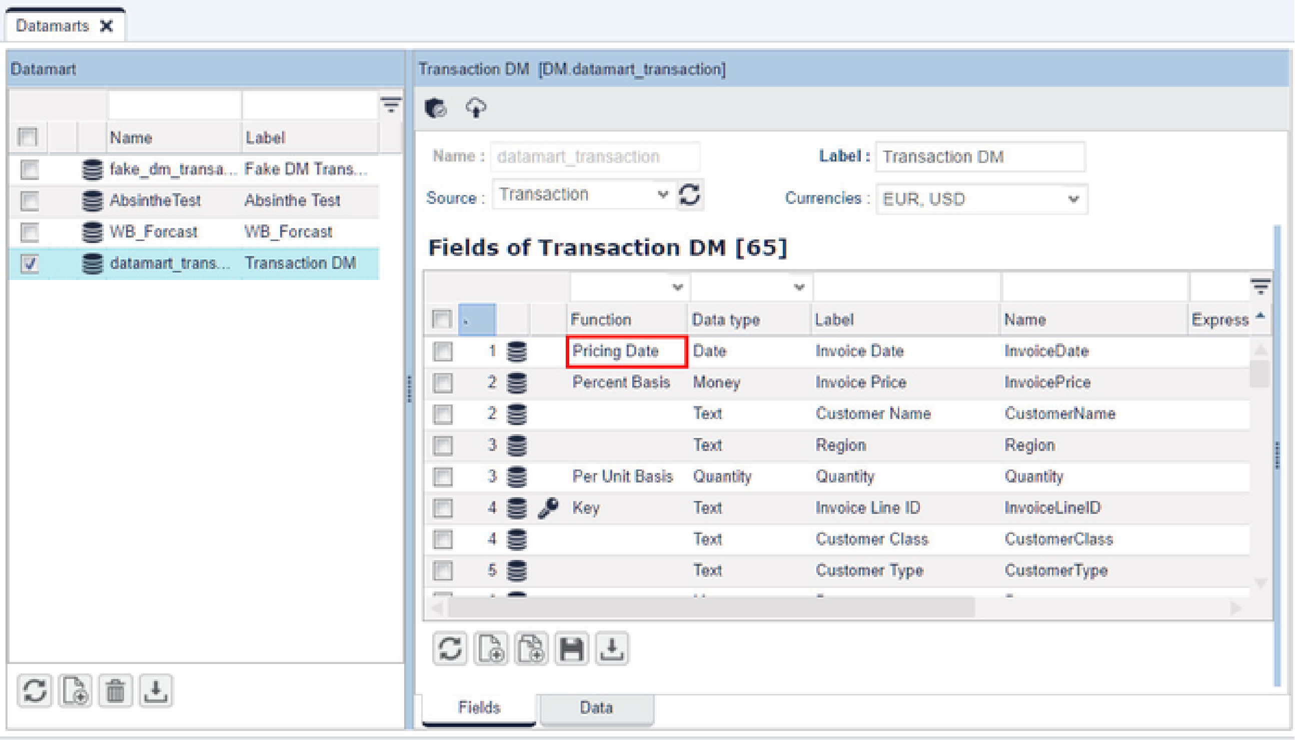
Name	Label
fake_dm_transa...	Fake DM Trans...
AbsintheTest	Absinthe Test
WB_Forcast	WB_Forcast
☑ datamart_trans...	Transaction DM

Name : datamart_transaction
Label : Transaction DM
Source : Transaction
Currencies : EUR, USD

Fields of Transaction DM [65]

	Function	Data type	...	...	...	Ov
1	Pricing Date	Date	I...	I...		Tra
2	Percent Basis	Money	I...	I...		Tra
2		Text	...	...		Cu
3	Per Unit Basis	Quantity	...	...		Tra
3		Text	...	...		Cu
4	Key	Text	I...	I...		Tra
4		Text	...	...		Cu
5		Text	...	...		Cu

4. Set the 'Function' of the date associated with the currency exchange rate to 'Pricing Date'. This will convert all Money fields in the Datamart to the query's target currency (which will be set later in the setOptions function).



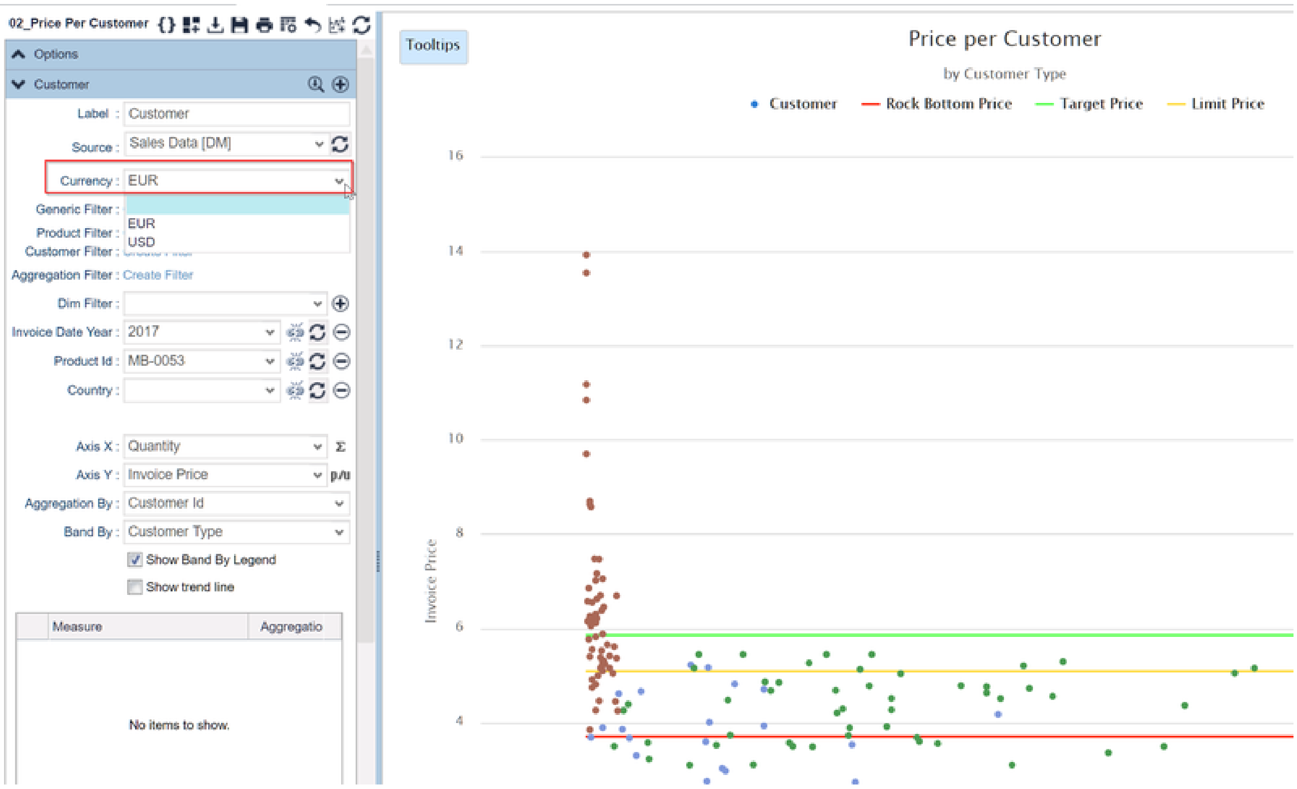
Name	Label
fake_dm_transa...	Fake DM Trans...
AbsintheTest	Absinthe Test
WB_Forcast	WB_Forcast
☑ datamart_trans...	Transaction DM

Name : datamart_transaction
Label : Transaction DM
Source : Transaction
Currencies : EUR, USD

Fields of Transaction DM [65]

.	Function	Data type	Label	Name	Express
1	Pricing Date	Date	Invoice Date	InvoiceDate	
2	Percent Basis	Money	Invoice Price	InvoicePrice	
2		Text	Customer Name	CustomerName	
3		Text	Region	Region	
3	Per Unit Basis	Quantity	Quantity	Quantity	
4	Key	Text	Invoice Line ID	InvoiceLineID	
4		Text	Customer Class	CustomerClass	
5		Text	Customer Type	CustomerType	

5. Currency conversion takes place in Datamart queries, in Groovy logic or PA charts and data tables/grids.
To use the currency in a Groovy logic, assuming you have `def query = dmCtx.newQuery(...)`, do this: `query.setOptions(["currency":"USD"]`
In the Charts UI, a currency can be selected from a drop-down:



02_Price Per Customer

Options

Customer

Label : Customer

Source : Sales Data [DM]

Currency : EUR

Generic Filter :

Product Filter :

Customer Filter :

EUR
USD

Aggregation Filter : Create Filter

Dim Filter :

Invoice Date Year : 2017

Product Id : MB-0053

Country :

Axis X : Quantity

Axis Y : Invoice Price

Aggregation By : Customer Id

Band By : Customer Type

☑ Show Band By Legend

□ Show trend line

Measure	Aggregatio
No items to show.	

The available currencies are sourced from the CcyTo column in the Currencies DataSource.

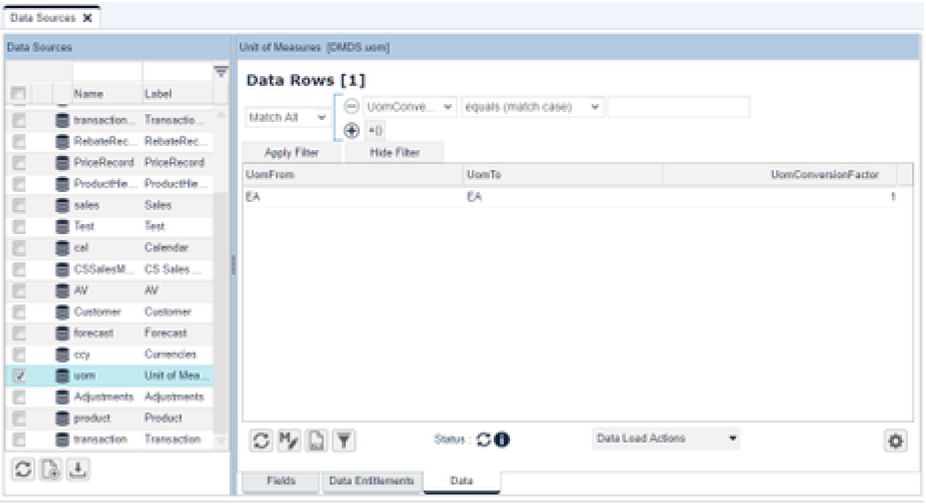
✓ In the application, the first currency shown in a drop-down menu is the default currency set in Datamart or in the partition settings (in this order). If none of these is set, the first currency is by alphabetical order.

ℹ Problem seeing prices ? Try this Q/A.

How to Load Units of Measure

Here is the process for loading Units of Measure (UOM) data into a Datamart using the UOM Data Source:

1. Set the data type of the UOM field in the Data Source to ‘UOM’. There can only be one UOM field in a Data Source. In a Data Source, the UOM field has an important meaning: the quantity field in that Data Source is expressed in the given UOM. (In a Datamart, you can have as many UOMs as you need because it is just like any other field in a Datamart - a field that you can query.)
2. Load data into the UOM Data Source, indicating which two UOMs you want to convert between and what the conversion rate is in the three fields. The UomTo field should always be your partition's Base UOM. You can identify the Base UOM by looking at the UOM-rate field. For example, if that field is 'VOLUME LOCAL UOM CDToEARate', then 'EA' is the Base UOM.



Data Sources

Data Sources

	Name	Label
□	transaction...	Transactio...
□	RebateRec...	RebateRec...
□	PriceRecord	PriceRecord
□	ProductHe...	ProductHe...
□	sales	Sales
□	Test	Test
□	cal	Calendar
□	CSSalesM...	CS Sales ...
□	AV	AV
□	Customer	Customer
□	forecast	Forecast
□	ccy	Currencies
☑	uom	Unit of Mea...
□	Adjustments	Adjustments
□	product	Product
□	transaction	Transaction

Unit of Measures [DMDS uom]

Data Rows [1]

Match All · UomConve... · equals (match case)

Apply Filter · Hide Filter

UomFrom	UomTo	UomConversionFactor
EA	EA	1

Status :

Data Load Actions

Fields · Data Entitlements · Data

3. When all of the UOM data are in the UOM DS, reload the Datamart.
    - If you do this with an empty Datamart, then you should see the UOM-rate field(s) in the Datamart populated.
4. If you already have data loaded into your Datamart, you must redeploy the Datamart to create the UOM-rate field(s) (you may need to log

out and log back in or just wait a bit for the fields to show up). After that, you can:

- Either completely reload your Datamart (via truncate and refresh, or refresh without the incremental checkbox checked).
- Or you could refresh the Datamart after setting the UOM Data Source as the source in the refresh Data Load. When you set a source, you 'force' all that source data in the Datamart, disregarding the incremental load behavior.

No columns are converted in the Datamart when the UOM-rate field is created, and the only way to use that field is in a Datamart query. When you run a query on a Datamart, all quantities are converted to the Base UOM. The Base UOM and Base CCY are configured on a partition level and are set in PartitionAdmin, which is done by whoever sets up the partition.

How to Build a New Excel Template (for Quote)

1. Navigate to a list of quotes, select one, and download it as an Excel file (see Download Quote to Excel or PDF for details).
2. Examine the Excel file – it has the "Data" sheet (line details of the quote), "Header" sheet (details of the Quote Header) and possibly other sheets too.
3. Create a new sheet with some fancy graphics where you leverage the data from the "Data" and "Header" sheets (simply use the Excel formulas to access the data from the sheets with data). For inspiration see the attached Excel example.
4. Upload this new Excel file into Pricefx (as described in Publishing Templates).
5. Try downloading a quote into Excel again – now you should get your Excel with the fancy graphics.

How to Create a Matrix Price List

A matrix price list is another type of price list. Through the addition of a second key dimension, prices can be calculated, displayed and maintained (multiple times) per combination of the SKU (product) and any other dimension, such as country, quantity, customer, etc.

1. Define a matrix pricing logic:
   a. Go to Configuration > Calculation Logic > Generic Logic and create a new one.
   b. Set the logic's Nature to Matrix.
   c. Define the logic's elements, including an element that is visible and returns a set of keys:

```
return [ "A", "B", "C" ]
```

2. Define a price list logic (or a product logic) that will reference the matrix logic and retrieve the key list:
   a. Go to Configuration > Calculation Logic > Generic Logic and create a new one.
   b. In the element, in which you need to use the value of the secondary key, you can retrieve it this way:

```
api.getSecondaryKey()
```

3. Create a price list:
   a. Go to PriceBuilder > Price Lists and create a new one.
   b. In Default pricing logic, select the price list logic created previously.
   c. In Matrix logic, select the matrix logic created previously.
   d. In Matrix logic element, select the element in the matrix logic that returns the key list.

How to Write a Logic in Groovy

BEFORE YOU BEGIN

You need to know:

- How to write a calculation logic in Formulas (at least the principles of calculation logics in Pricefx).
- You have some experience with object oriented programming (regardless of the programming language).
- The syntax of the Groovy language – there are many resources on the Internet, and for Pricefx Groovy functions, we have API Documentation and in this video we will show you how to change the logic written in Formulas into Groovy.

CONCEPT

There are over 150 functions that you can use directly when you type "api." and the name of the function. For example:

- api.getElement("Total_Sales")
- api.trace("Current month - Last month", currentMonth, lastMonth)
- api.find("P", Filter.equals("attribute1","Beef"))

Writing a logic basically means to write a code in Groovy and call these Pricefx specific functions.

FUNCTIONS DOCUMENTATION

- Functions sorted by categories of usage with some examples and screenshots: Groovy API Functions

- Complete Pricefx Groovy API with all objects and the syntax of all functions: https://developer.pricefx.eu/api/groovy/master/net/pricefx/formulaengine/scripting/PublicGroovyAPI.html

EXAMPLES

Example 1. of conversion logic from Formulas to Groovy

Element	Formula	Groovy

Cost	`Avg(`
	`  Column(`
	`    PXLookup("ProductCost"),`
	`  "Cost"`
	`  )`
	`)`
		`def sum = 0.0`
		`def costRows =`
		`api.productExtension("Produc`
		`for (row in costRows) sum +=`
		`row["attribute1"]`
		`return sum / costRows.size()`

ProductGroup	`Product("attribute1")`	`api.product("attribute1")`

MarginAdj	`VLookup("MarginAd",`
	`ProductGroup)`
		`api.vLookup("MarginAd",`
		`api.getElement("ProductGroup`

MarginEuro	`Cost * MarginAdj`	`api.getElement("Cost") *`
		`api.getElement("MarginAdj")`

Quality	`Product("attribute7")`	`api.product("attribute7")`

AttributeAdj	`VLookup("Quality", Quality)`
	`* Cost`
		`api.vLookup("Quality",`
		`api.getElement("Quality")) *`
		`api.getElement("Cost")`

ListPrice	`Cost + MarginEuro +`
	`AttributeAdj`
		`api.getElement("Cost") +`
		`api.getElement("MarginEuro")`
		`api.getElement("AttributeAdj`

Example 2. of conversion logic from Formulas to Groovy

Element	Formula	Groovy

ListPrice	`Pricelist("Global`
	`Pricelist 2017")`
		`api.pricelist("Global P`

SalesDiscountPercentage	`UserEntry("Sales dicount`
	`(%)") / 100`
		`api.userEntry("Sales di`

Quantity	`UserEntry("Quantity")`	`api.userEntry("Quantity`

SalesDiscount	`ListPrice`
	`* SalesDiscountPercentage`
		`api.getElement("ListPri`
		`api.getElement("SalesDi`

InvoicePrice	`ListPrice - SalesDiscount`	`api.getElement("ListPri`
		`- api.getElement("SalesD`

How to Configure Change History for PM or QC

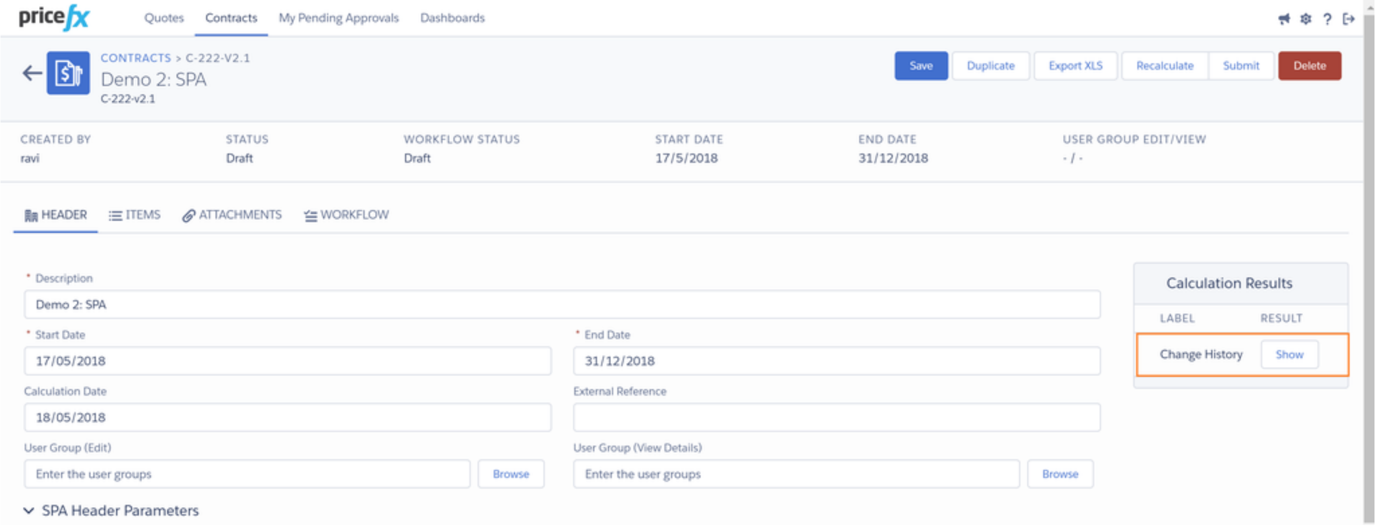
This article explains how to show changes between current and previous revisions of a Contract in PromotionManager. In this particular case we had created two different contract term types: By Bulletin and By Product. So a Contract can have line items of the Bulletin or Product type. In order to configure the Change History tool, we need to define the Change History element in Configuration > Promotion Manager > Contract Header Logics. Here is the Groovy code that will display the Contract History element in Contract header output results.

```
def roundUp(numberToRound,roundTo)
{
      return Math.ceil(numberToRound/roundTo)*roundTo
```

```
}

def changeHistoryMatrix = api.newMatrix("Bulletin", "Product", "Previous
Quantity", "Current Quantity", "Previous Into Stock Multiplier",
"Current Into Stock Multiplier", "Previous Requested Multiplier",
"Current Requested Multiplier", "Previous Gross Margin %", "Current
Gross Margin %" )

def view = cProcessor.getContractView()
def prevRevUniqueName = view.prevRev

def prevRevision = api.find("CT", Filter.equal("uniqueName",
prevRevUniqueName))
if (!prevRevision) {
     return
}

def prevLineItemsCollection =
api.getCalculableLineItemCollection(prevRevision[0].typedId)
def prevLineItems = prevLineItemsCollection?.lineItems

for (line in view.lineItems) {
     if (line.folder) continue

     def currentBulletin = line.outputs?.find { it.resultName ==
"Bulletin" }?.result
     def currentProduct = line.outputs?.find { it.resultName ==
"Product" }?.result
     def currentQuantity = line.outputs?.find { it.resultName ==
"Quantity" }?.result
     def currentIntoStockMultiplier = line.outputs?.find { it.resultName
== "IntoStockMultiplier" }?.result
     def currentRequestedMultiplier = line.outputs?.find { it.resultName
== "RequestedMultiplier" }?.result
     def currentGrossMargin = line.outputs?.find { it.resultName ==
"GrossMarginPercent" }?.result
     def currentGrossMarginPercent = roundUp((currentGrossMargin *
100),2) + " %"

     for (prevLine in prevLineItems) {
         if (prevLine.folder) continue

         def prevBulletin = prevLine.outputs?.find { it.resultName ==
"Bulletin" }?.result
         def prevProduct = prevLine.outputs?.find { it.resultName ==
"Product" }?.result
         def prevQuantity = prevLine.outputs?.find { it.resultName ==
"Quantity" }?.result
         def prevIntoStockMultiplier = prevLine.outputs?.find {
it.resultName == "IntoStockMultiplier" }?.result
```

```
         def prevRequestedMultiplier = prevLine.outputs?.find {
it.resultName == "RequestedMultiplier" }?.result
         def prevGrossMargin = prevLine.outputs?.find { it.resultName ==
"GrossMarginPercent" }?.result
         def prevGrossMarginPercent = roundUp((prevGrossMargin * 100),2)
+ " %"

         if (prevBulletin == currentBulletin && prevProduct ==
currentProduct) {
              changeHistoryMatrix.addRow([
                  "Bulletin" : currentBulletin,
                  "Product" : currentProduct,
                  "Previous Quantity" : prevQuantity,
                  "Current Quantity" : currentQuantity,
                  "Previous Into Stock Multiplier" :
prevIntoStockMultiplier,
                  "Current Into Stock Multiplier" :
currentIntoStockMultiplier,
                  "Previous Requested Multiplier" :
prevRequestedMultiplier,
                  "Current Requested Multiplier" :
currentRequestedMultiplier,
                  "Previous Gross Margin %" : prevGrossMarginPercent,
                  "Current Gross Margin %" : currentGrossMarginPercent

              ])
         }
    }
}

cProcessor.addOrUpdateOutput (
        "ROOT",
        [  "resultName" : "ChangeHistory",
           "resultLabel": "Change History",
           "result" : changeHistoryMatrix,
                "resultType" : "MATRIX"
            ]
   )

```

Contract Calculation Results (Unity UI)



CONTRACTS > C-222-V2.1
Demo 2: SPA
C-222-v2.1

CREATED BY	STATUS	WORKFLOW STATUS	START DATE	END DATE	USER GROUP EDIT/VIEW
ravi	Draft	Draft	17/5/2018	31/12/2018	- / -

* Description: Demo 2: SPA

* Start Date: 17/05/2018

* End Date: 31/12/2018

Calculation Date: 18/05/2018

External Reference:

User Group (Edit): Enter the user groups

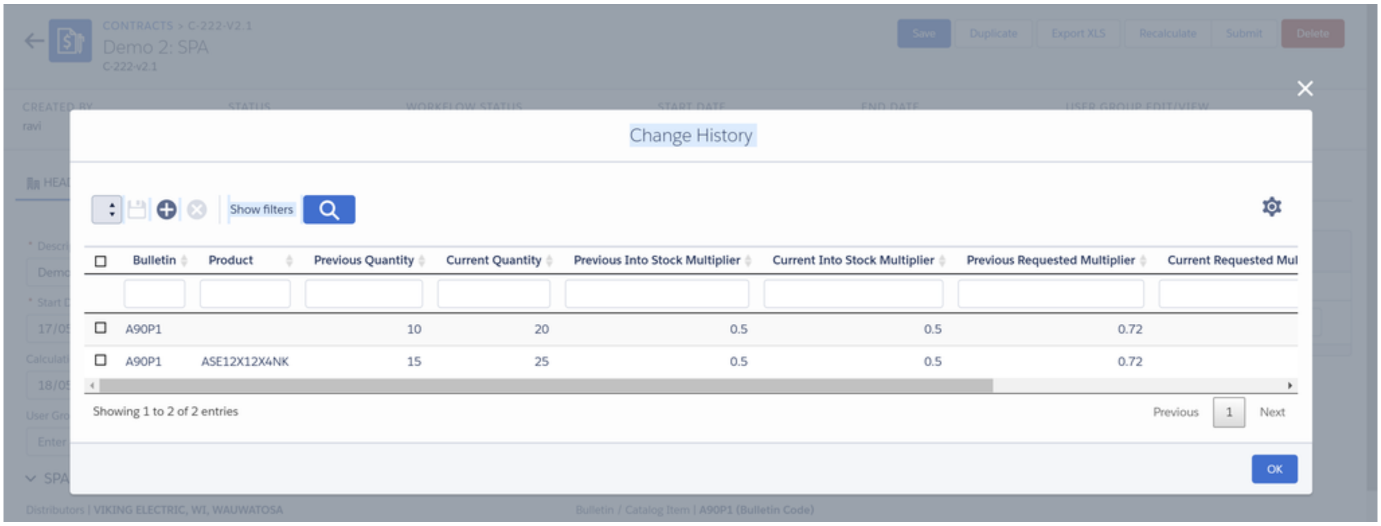
User Group (View Details): Enter the user groups

SPA Header Parameters

Calculation Results

LABEL	RESULT
Change History	Show

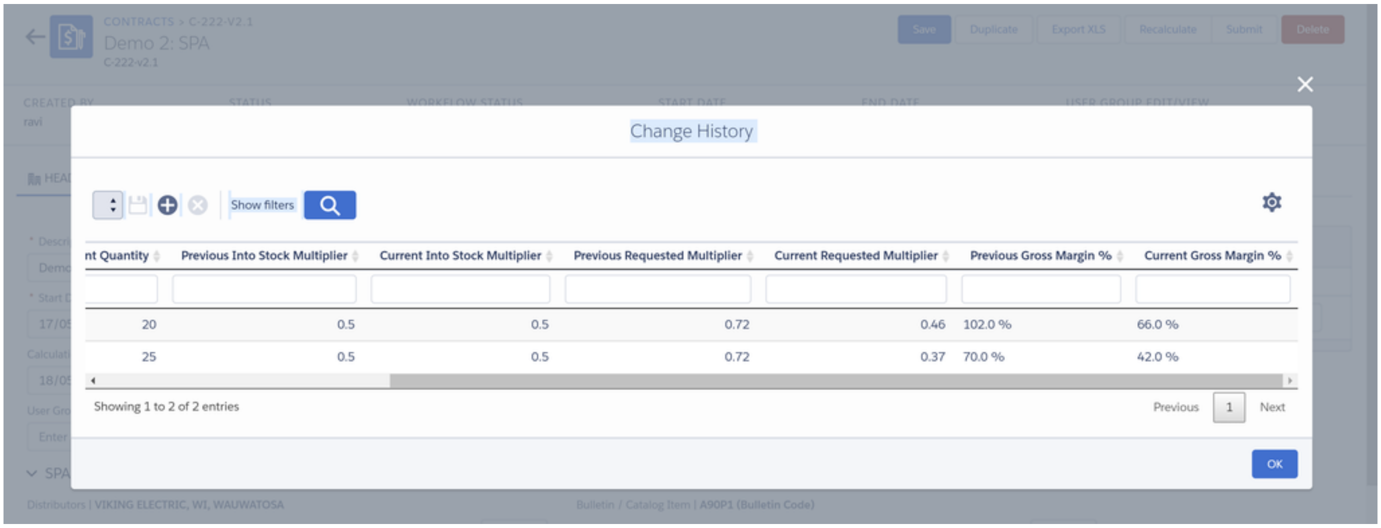
Change History (Unity UI)



Change History

☐	Bulletin	Product	Previous Quantity	Current Quantity	Previous Into Stock Multiplier	Current Into Stock Multiplier	Previous Requested Multiplier	Current Requested Mul
☐	A90P1		10	20	0.5	0.5	0.72	
☐	A90P1	ASE12X12X4NK	15	25	0.5	0.5	0.72	

Showing 1 to 2 of 2 entries



Change History

nt Quantity	Previous Into Stock Multiplier	Current Into Stock Multiplier	Previous Requested Multiplier	Current Requested Multiplier	Previous Gross Margin %	Current Gross Margin %
20	0.5	0.5	0.72	0.46	102.0 %	66.0 %
25	0.5	0.5	0.72	0.37	70.0 %	42.0 %

Showing 1 to 2 of 2 entries

Advanced Usage Tips

- How to Query a Data Source Directly
- How to Transfer All Data at Once between Two Partitions
- How to Automatically Archive Old Data
- How to (Re)Create Missing Default PA Data Loads
- How to Import PX Data to a Datamart
- How to Work with Inputs in PA Formulas
- How to Create Revenue Waterfall Dashboard
- How to Add a Gauge Chart to Line Detail and Header
- How to Add a Matrix to Quote Header Output Fields
- How to Add a Quote Link to Dashboard
- How to Debug a Price List
- How to Add Entry to Matrix PP
- How to Install .NET 3.5 (Required for Excel Client) with Windows 10
- How to Run Excel Client on Mac
- How to Find Set of Quotes Containing Given Products
- How to Generate Excel Reports from Calculation Logic
- How to Reference Values Between Two Price Parameter Columns
- How to Populate Customer or Product Picker Created via API
- How to Store Quote Inputs/Outputs in PriceAnalyzer
- How to Add Inputs to Product Selector
- How to Add Approver to Workflow Email Template
- How to Create Parametric Approval Workflow
- How to Create Button in Email Template
- How to Use Secondary Key to Create Matrix Price Lists
- How to Create Cascading Filters
- How to Create Portlet Alternating Between Chart and HTML
- How to Create Quote Line Items from Header Logic
- How To Generate Approval Required E-mail On Quote Save
- How To Disable Submit Quote Button in QuoteConfigurator
- How to Hide, Disable or Make Required Fields/Buttons
- How to Update Object References Regularly
- How to Set Up a Buy-One-Get-One-Free Promotion
- How to Automatically Organize Products into Folders
- How to Set Data with Null Values in FlexCharts?
- How to Get Quote's Effective and Expiry Dates using Calculation Logic
- FlexChart Examples
- How to Add Your Own Button to Navigation
- Stacked Column Chart (using the Flex Chart)
- How to Set Value Labels in Drop-Down in Header Logic
- How to Access QC Line Outputs in Workflow Template
- How to Display Large Number of Data Points in FlexChart Scatter Plot
- How to Add a Map to Dashboard
- How to Delete Deployed Calculation Flow
- How to Pass Header Input Values to Inline Configurator
- How to Mass Update Quotes with a Configurator
- How to Make Home Panels Always Visible
- How to Deal with PA Data Table's UI Responsiveness?
- How to Get Full Log File for Specific Node
- How to Make the Simple Filter Search for Exact Values

- How to Hide Download DOCX Icon
- How to Add Image to Product Detail
- How to Set Language and Locale via URL
- How to identify whether a quote is duplicated, or revised, or new in Quote Header Logic?
- Inputs Behavior in Quote Revisions
- How to Configure Segmentation and Optimization Models
- How to See Users Assigned to a User Group
- How to Disable Autocomplete When Entering Values in Advanced Filters
- How to Set Up SSO with Azure

How to Query a Data Source Directly

It is possible to query data straight from a data source, instead of loading it into a Datamart and querying from there. This can save unnecessary processing of loading from DS to DM.

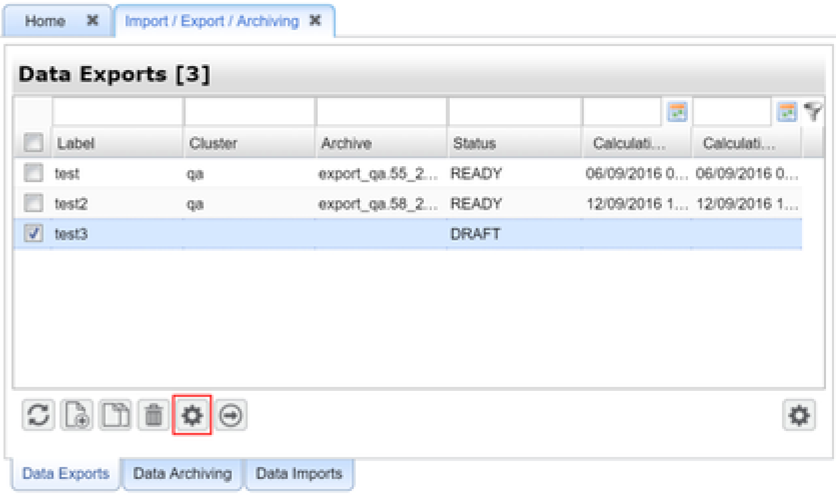
Refer to the following API links for more in-depth info:

- DatamartContext.getDataSource()
- DatamartContext.executeQuery()

How to Transfer All Data at Once between Two Partitions

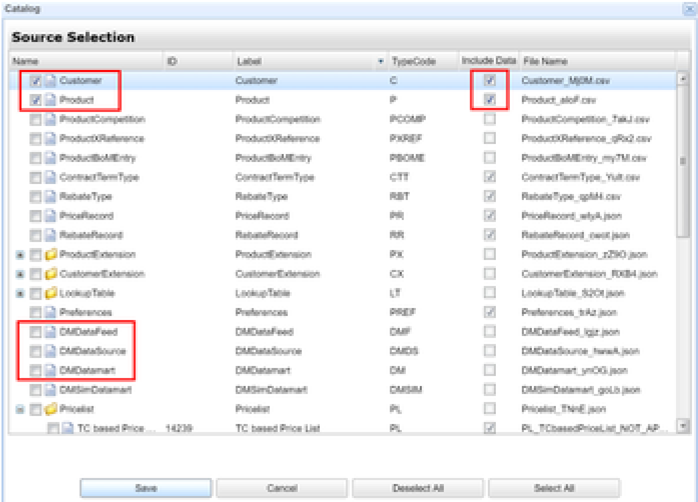
To transfer all data at once between the two partitions/environments, follow these steps:

1. In the Admin section, go to Import / Export / Archiving.
2. Click the Add icon.
3. Provide a label for your new data export item.
4. Select your data export item and click the Configure icon.



	Label	Cluster	Archive	Status	Calculati...	Calculati...
□	test	qa	export_qa.55_2...	READY	06/09/2016 0...	06/09/2016 0...
□	test2	qa	export_qa.58_2...	READY	12/09/2016 1...	12/09/2016 1...
☑	test3			DRAFT		

5. In the Source Selection window, select what sources should be exported.
   Notes:
   - The Name column tells you what object structure will be exported.
   - The Include Data check-box determines if the actual data will be exported.
   - The sources named DMDataxxx represent PA data (e.g. transactions). If these sources are selected, the export may take a very long time and the resulting file will be huge. It is not mandatory to have these items selected. If you don't need this data, make sure the check-boxes are not checked.
   - Note what fields are selected – later in Step 15 the same fields must be checked/unchecked, otherwise the import will fail.

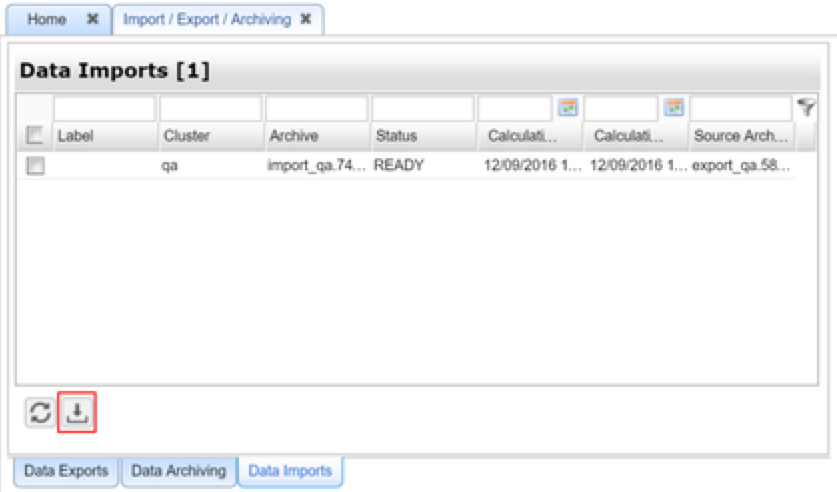


Catalog

Source Selection

Name	ID	Label	TypeCode	Include Data	File Name
☑ Customer		Customer	C	☑	Customer_Mj0M.csv
☑ Product		Product	P	☑	Product_aloF.csv
□ ProductCompetition		ProductCompetition	PCOMP	□	ProductCompetition_TakJ.csv
□ ProductXReference		ProductXReference	PXREF	□	ProductXReference_gRx2.csv
□ ProductBoMEntry		ProductBoMEntry	PBOME	□	ProductBoMEntry_myTM.csv
□ ContractTermType		ContractTermType	CTT	☑	ContractTermType_YuIt.csv
□ RebateType		RebateType	RBT	☑	RebateType_gpM4.csv
□ PriceRecord		PriceRecord	PR	☑	PriceRecord_wtyA.json
□ RebateRecord		RebateRecord	RR	☑	RebateRecord_owot.json
□ ProductExtension		ProductExtension	PX	□	ProductExtension_zZ9O.json
□ CustomerExtension		CustomerExtension	CX	□	CustomerExtension_RXB4.json
□ LookupTable		LookupTable	LT	□	LookupTable_S2Ot.json
□ Preferences		Preferences	PREF	☑	Preferences_trAz.json
□ DMDataFeed		DMDataFeed	DMF	□	DMDataFeed_lgjz.json
□ DMDataSource		DMDataSource	DMDS	□	DMDataSource_hwwA.json
□ DMDatamart		DMDatamart	DM	□	DMDatamart_ynOG.json
□ DMSimDatamart		DMSimDatamart	DMSIM	□	DMSimDatamart_goLb.json
□ Pricelist		Pricelist	PL	□	Pricelist_TNnE.json
□ TC based Price ...	14239	TC based Price List	PL	☑	PL_TCbasedPriceList_NOT_AP...

6. Click Save and the Source Selection window closes.
7. Click the Start ➡ icon.
8. Wait until the status of your data export changes from Pending to Ready. (Click the Refresh icon to update the status.)
9. Download the Download archive icon and save the resulting file locally.
10. Log in to the partition where you want to import the data.
11. In the Admin section, go to Import / Export / Archiving.
12. Switch to the Data Imports sub-tab.
13. Click the Upload icon, browse for your downloaded file and confirm. The upload starts.



Data Imports [1]

□	Label	Cluster	Archive	Status	Calculati...	Calculati...	Source Arch...
□		qa	import_qa.74...	READY	12/09/2016 1...	12/09/2016 1...	export_qa.58...

14. On the Data Imports sub-tab, select your data import item (which just got created by the upload) and click the Configure icon.
15. In the Source Selection window, select what sources and data should be imported.

   ⚠ Make sure the same fields are checked/unchecked as during export in Step 5, otherwise the import will fail.
16. Click Save and the Source Selection window closes.
17. Click the Start icon.
18. Wait until the status of your data import changes from Pending to Ready. (Click the Refresh icon to update the status.

How to Automatically Archive Old Data

ℹ Starting archiving tasks from a calculation flow is available as of the Mai Tai release.

Scenario

A customer has several millions of rows in the Product Competition table (this can easily happen when loading data from companies who gather data from many competitors). But you're using only data from the last month in this table.

You can use the Archiving utility (Configuration > System > Import / Export / Archiving) but this is a manual process. How to do this automatically at regular intervals?

Solution

1. Create an "Archiving Template" – i.e. prepare the Archiving process (the same as manual), but do not run it.

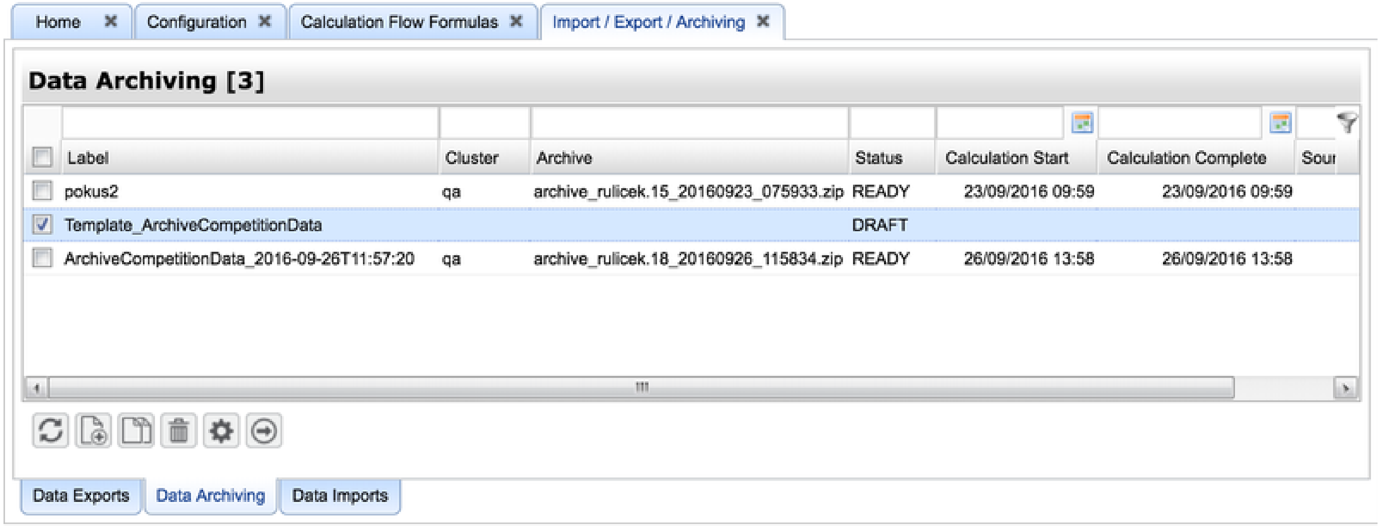
2. Create a Calculation Flow Formula (CFF) which will execute the Archiving process.
3. Create a Calculation Flow – i.e. set the scheduler to run the CFF every month.

Example of the Calculation Flow Formula

```
def date = (new Date()).format("yyyy-MM-dd'T'HH:mm:ss")
def archiveActionSourceTemplate = "Template_ArchiveCompetitionData"
//label of the manually created Archiving process
def archiveActionResultName = "ArchiveCompetitionData_" + date

actionBuilder
  .addExportAction(archiveActionResultName)
    .doArchive(archiveActionSourceTemplate)
```

Example

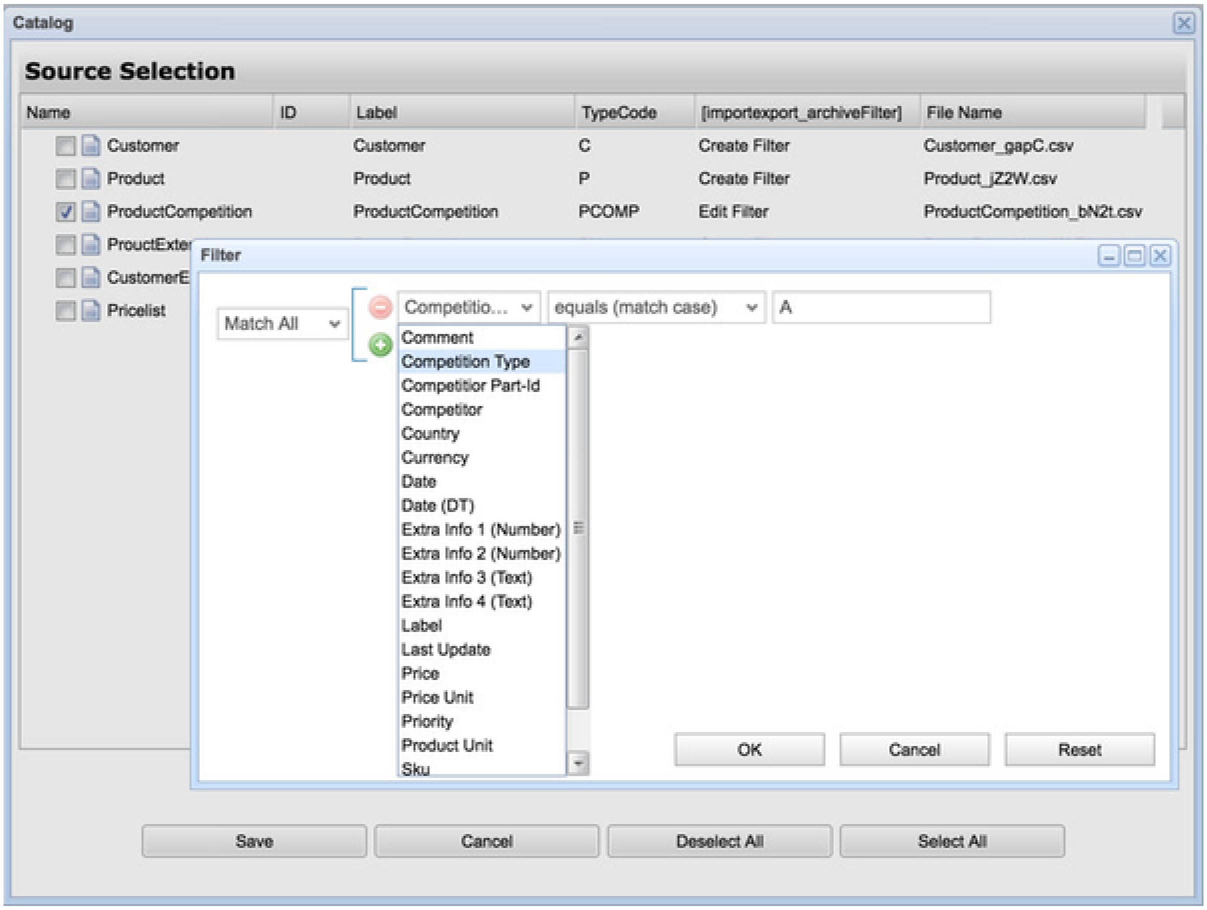
The second item in the table shows the Template, the third one shows the result.



Data Archiving [3]

□	Label	Cluster	Archive	Status	Calculation Start	Calculation Complete	Sour
□	pokus2	qa	archive_rulicek.15_20160923_075933.zip	READY	23/09/2016 09:59	23/09/2016 09:59	
☑	Template_ArchiveCompetitionData			DRAFT			
□	ArchiveCompetitionData_2016-09-26T11:57:20	qa	archive_rulicek.18_20160926_115834.zip	READY	26/09/2016 13:58	26/09/2016 13:58	

This is the configuration of the Template in detail:



Catalog

Source Selection

Name	ID	Label	TypeCode	[importexport_archiveFilter]	File Name
□ Customer		Customer	C	Create Filter	Customer_gapC.csv
□ Product		Product	P	Create Filter	Product_jZ2W.csv
☑ ProductCompetition		ProductCompetition	PCOMP	Edit Filter	ProductCompetition_bN2t.csv
□ ProuctExter					
□ CustomerE					
□ Pricelist					

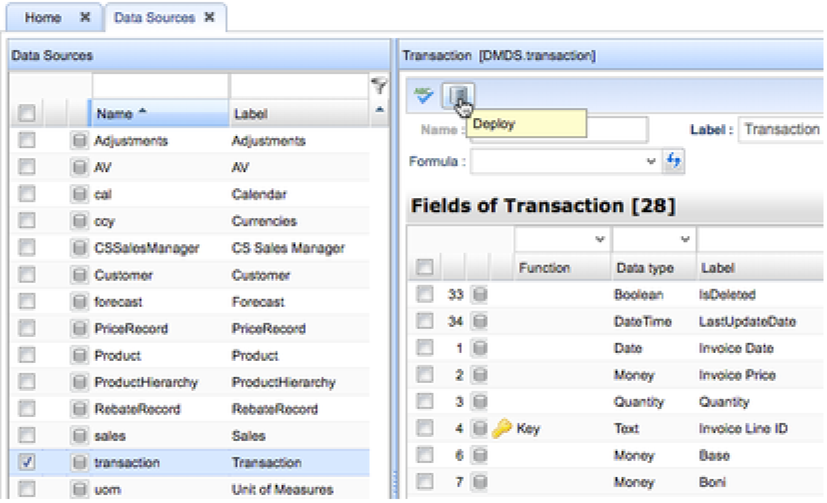
Filter: Match All; Competitio… equals (match case) A

How to (Re)Create Missing Default PA Data Loads

In PriceAnalyzer, for each Data Feed, Data Source and Datamart there should be a couple of Data Loads created automatically. If any of those are missing, try the following solutions.

Solution 1 – Deploy

Deploying (or re-deploying) the Data Source or Datamart should create the default Data Loads. If this does not help, try Solution 2.



Data Sources

	Name	Label
□	Adjustments	Adjustments
□	AV	AV
□	cal	Calendar
□	ccy	Currencies
□	CSSalesManager	CS Sales Manager
□	Customer	Customer
□	forecast	Forecast
□	PriceRecord	PriceRecord
□	Product	Product
□	ProductHierarchy	ProductHierarchy
□	RebateRecord	RebateRecord
□	sales	Sales
☑	transaction	Transaction
□	uom	Unit of Measures

Transaction [DMDS.transaction]

Name : Label : Transaction

Formula :

Fields of Transaction [28]

		Function	Data type	Label
□	33		Boolean	IsDeleted
□	34		DateTime	LastUpdateDate
□	1		Date	Invoice Date
□	2		Money	Invoice Price
□	3		Quantity	Quantity
□	4	Key	Text	Invoice Line ID
□	6		Money	Base
□	7		Money	Boni

Solution 2 – System Command

If the missing Data Loads are related to the default Data Sources and Data Feeds (Currencies, Calendar, Customer, Product, UOM etc.), ask Support to run a system redeploy command for you.

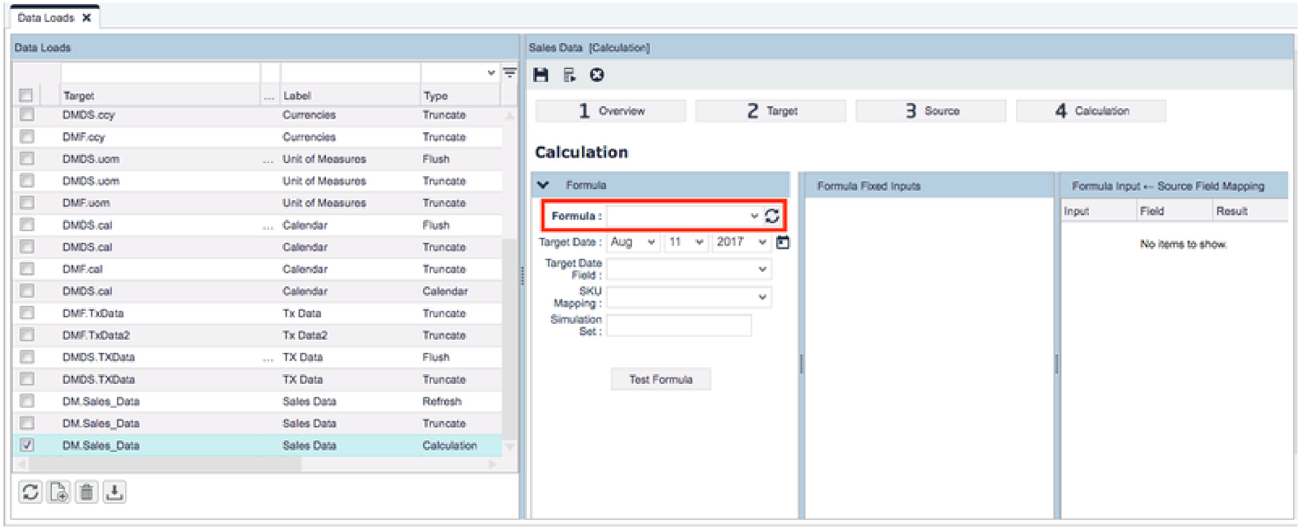
How to Import PX Data to a Datamart

We can import PX data to the Datamart using a PA Calculation Logic:

1. Create a PA Calculation Logic with a Groovy element (set the Calculation Context of this element to Init) similar to this "Competition" PX Category example:

```
def target = api.getDatamartRowSet("target")
def result = api.stream("PX", null /* sort-by*/,

["sku","attribute1","attribute2","attribute5","attribute7","attribu
te9","attribute11"],
                        Filter.equal("name", "Competition"))

while (result.hasNext()) {
   def comp = result.next()
   // to be able to load rows in the DS we need at least the key
fields
   if (comp.sku != null && comp.attribute1 != null) {
      def row = ["DateTime":now,
                 "sku": comp.sku,
                 "Competitor" : comp.attribute1,
                 "Price" : comp.attribute2,
                 "ShippingCost" : comp.attribute5,
                 "DataProvider" : comp.attribute7,
                 "Availability" : comp.attribute9,
                 "Relevant" : comp.attribute11]
      target.addRow(row)
   }
}
result.close()
```

2. Provided you already created an appropriate Data Source to store the data, in the Data Loads tab you can add a Data Load of the type "Calculation" which uses the above logic to load your PX data.



□	Target	...	Label	Type
□	DMDS.ccy		Currencies	Truncate
□	DMF.ccy		Currencies	Truncate
□	DMDS.uom	...	Unit of Measures	Flush
□	DMDS.uom		Unit of Measures	Truncate
□	DMF.uom		Unit of Measures	Truncate
□	DMDS.cal	...	Calendar	Flush
□	DMDS.cal		Calendar	Truncate
□	DMF.cal		Calendar	Truncate
□	DMDS.cal		Calendar	Calendar
□	DMF.TxData		Tx Data	Truncate
□	DMF.TxData2		Tx Data2	Truncate
□	DMDS.TXData	...	TX Data	Flush
□	DMDS.TXData		TX Data	Truncate
□	DM.Sales_Data		Sales Data	Refresh
□	DM.Sales_Data		Sales Data	Truncate
☑	DM.Sales_Data		Sales Data	Calculation

Calculation

Formula :
Target Date : Aug 11 2017
Target Date Field :
SKU Mapping :
Simulation Set :

Input	Field	Result

No items to show.

This next example is not related to the one before, it only shows in principle how to cache data read from PX table to a global Map

```
def sku2Competitor = api.global.sku2Competitor
if (sku2Competitor == null){
  sku2Competitor = [:]
  api.global.sku2Competitor = sku2Competitor
}
def skn = api.getElement("SKN")
if (sku2Competitor.containsKey(skn)){
  return sku2Competitor.get(skn)
}
else{
  // attribute4 = Competitor
  def pxNameFilter = api.filter("name", "NamedCompetition")
  def pxCompFilter = api.filter("attribute4",
api.getElement("Competitor"))
  def pxSKNFilter = api.filter("sku", api.getElement("SKN"))
  def competitors = api.find("PX", 0,1, null /*sort*/, ["id"],
pxNameFilter, pxCompFilter, pxSKNFilter)
  //api.trace("test", competitors?.size().toString(), pxCompFilter)
  def hasCompetitor = competitors?.size() > 0? 1:0
  sku2Competitor.put(skn, hasCompetitor)

  return hasCompetitor
}
```

A global variable 'sku2Competitor' is created to ensure that the logic is not run multiple times for the same SKN. In the else statement, we set up filters using `api.filter("fieldName", value)`. Then we create an array "competitors" using `api.find(String typeCode, int startRow, int maxRows, String sortBy,List<String> fields, Filter... filters)` with the filters that we created to find the row in the NamedCompetition PX that has the given Competitor and SKN. The SKN and Competitor are then stored in the global variable.

How to Work with Inputs in PA Formulas

To get field values to be used in formulas:

1. To obtain data from each row of a Datamart, use:

   ```
   UserEntry("fieldName")
   ```

   and

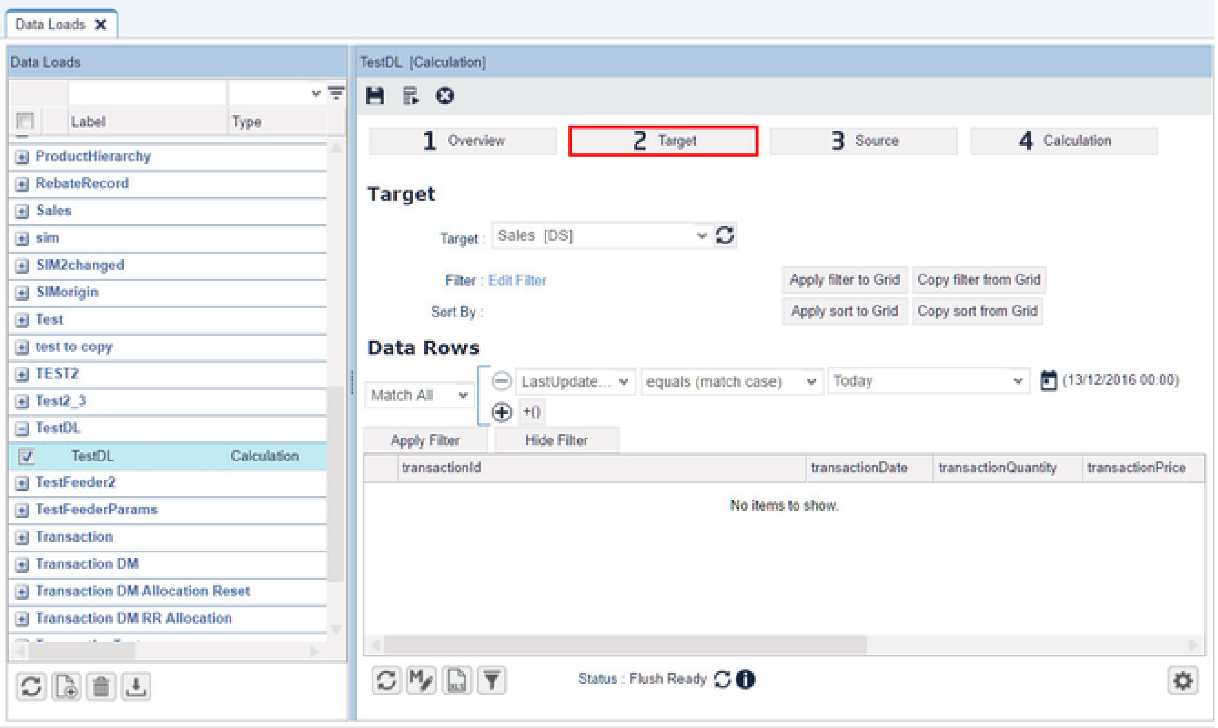
   ```
   StringUserEntry("fieldName")
   ```

2. After running the logic in a Data Load, as it iterates through each row of the Target Datamart, it will automatically access the Field Name parameter within the `UserEntry("fieldName")` formula. This field value can then be used in the rest of the logic.

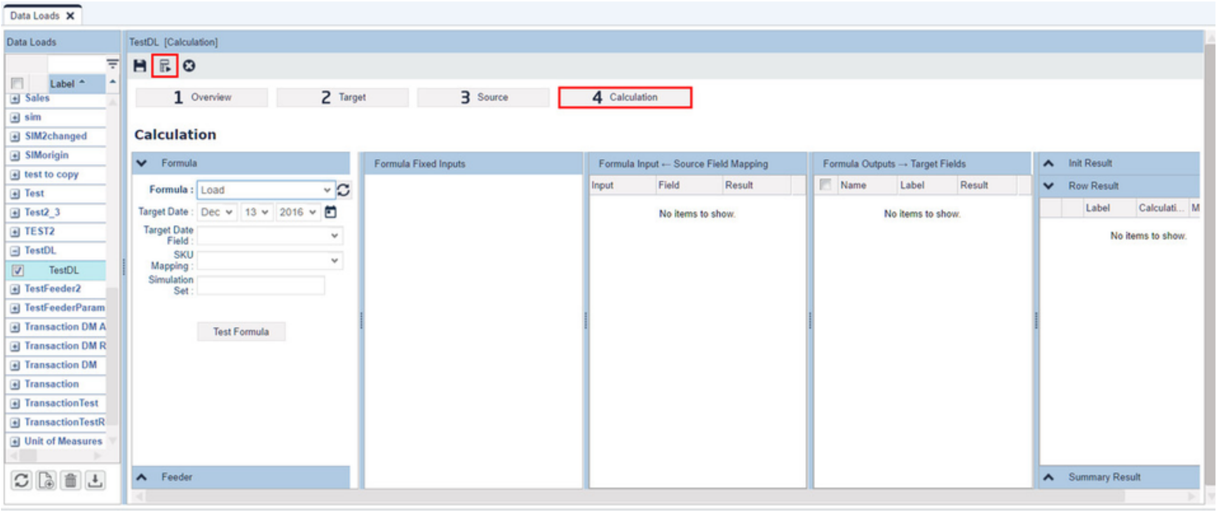
To add data to a field with a calculation logic:

1. Give an element in your logic the same name as the target field in the Datamart and whatever is returned in that element will be the input for that field in the Datamart when the logic is run.
2. To run the logic:

a. Create a new Data Load by clicking the 'Add' button and give it the same name as the target Datamart with the type 'Calculation'.
b. Go to the 'Target' tab and select a target Datamart from the drop-down menu. You can filter the Datamart here so the logic will only be run on certain rows of the Datamart.



c. Go to the '4 Calculation' step and select your formula from the drop-down menu. You will see your inputs in the 'Formula Fixed Inputs' column.
d. To test the formula, select a Datamart row in the 'Target' tab and click 'Test Formula'. You will see the results in the 'Formula Outputs' and 'Row Result' columns. You can click the Save button to store your filter on the Datamart and your formula.
e. Click 'Run Now' to run the logic on the target Datamart.



How to Create Revenue Waterfall Dashboard

Written by:	Akash Duseja

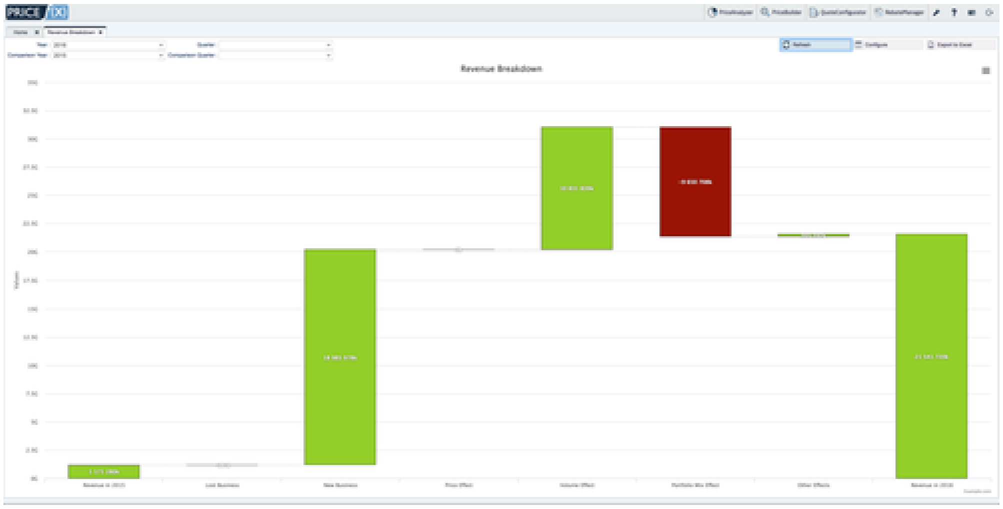
The following example shows how to implement a Revenue Waterfall Dashboard. It compares the year 2016 to 2015 but we can configure it to compare Quarters, Months, Weeks, Days etc.

The different components (from left to right) include:

- **Revenue in Time Period 1** – Total revenue in Time Period 1.
- **Lost Business** – Revenue from lost customers + Revenue from Retired Products + Revenue from current customers not buying

products in Time Period 2 (who did buy the same products in Time Period 1).

- **New Business** – Revenue from new customers + Revenue from New Products + Revenue from existing customers buying products in Time Period 2 (who didn't buy the same products in Time Period 1).
- **Price Effect** – Changes in total revenue due to changes in Price Per Unit of all products.
- **Volume Effect** – Changes in total revenue due to changes in volume of goods sold to customers.
- **Portfolio Mix Effect** – Changes in total revenue due to change in mix of sales (changes in proportions of a product's sales to total sales for time periods in question). For example, over a given time period, increase in share of a high-priced product will contribute to revenue positively and vice versa.
- **Other Effects** – Anything not accounted for by any of the above-mentioned effects.
- **Revenue in Time Period 2** – Total revenue in Time Period 2.

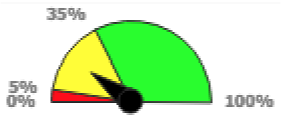


Technical details of the dashboard creation:

1. **Drop-down selections**: Use `DatamartContext.dimFilterEntry("Name", Column)` to get the drop-downs in the dashboard.
2. Create tables for time periods in question using `SelectQuery(..computed columns..).where(api.newDatamartSlice("DateColumnName", period))` for both periods. In our example, we obtain columns like Revenue, Quantity and PricePerUnit from these tables.
3. **Read revenues** from Time Periods just by obtaining `SUM(Revenue)` from the tables created in Step 2.
4. To get **Lost Business**, we use "`SELECT T1.* FROM T1 LEFT OUTER JOIN T2 ON T1.Product=T2.Product AND T1.Customer=T2.Customer WHERE T2.Product IS NULL`" where T1 is the comparison period and T2 is the current period (or the more recent of the two). Select Revenue from the result to get the Lost Business Amount.
5. To get **New Business**, we use "`SELECT T2.* FROM T1 RIGHT OUTER JOIN T2 ON T1.Product=T2.Product AND T1.Customer=T2.Customer WHERE T1.Product IS NULL`" where T1 is the comparison period and T2 is the current period (or the more recent of the two). Select Revenue from the result to get the New Business Amount.
6. Next, we fetch the business that does not fall in lost or new business category and we will call it **common business**. The easiest way to get that data would be to use the query "`SELECT * FROM T1 INNER JOIN T2 ON T1.Product=T2.Product AND T1.Customer=T2.Customer`".
7. To get the **Volume Effect**, we iterate through common business data and compute `((T2.Volume - T1.Volume) / T1.Volume)` to get the fraction increase in Volume and multiply that by T1.Revenue to get the volume effect for that row of data in the common business. Add that up over all the rows and we will have the total Volume Effect.
8. To get **Price Effect**, iterate through common business data and compute `(T2.PricePerUnit – T1.PricePerUnit)` and multiply that by T2.Volume. That will give us the change in revenue due to change in Price Per Unit for each row of common business. Now add this up for all the rows to compute the total Price Effect.
9. To calculate the **Portfolio Mix Effect**, iterate through common business data and compute `(T1.PricePerUnit) * (T2.Volume/SUM(T2.Volume) - T1.Volume/SUM(T1.Volume))` for every row. Multiplying the sum of these computed values to the `SUM(T2.Volume)` will give us the total Portfolio Mix Effect.
10. Plotting all of these computed items using a custom waterfall template will allow you to create a dashboard that will look similar to the example shown here.

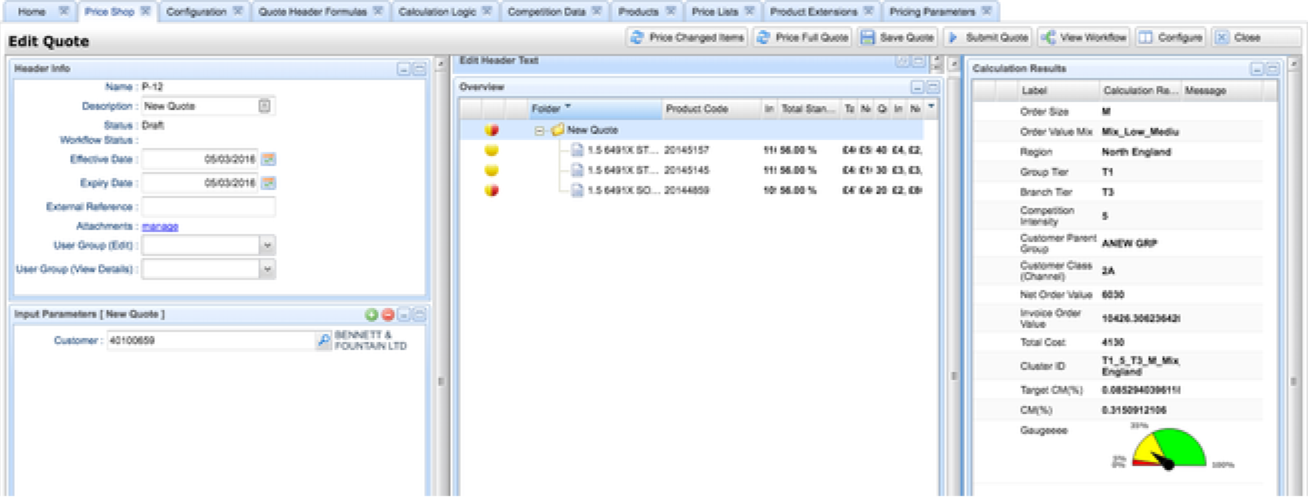
How to Add a Gauge Chart to Line Detail and Header

The Gauge chart is a convenient way to show, for example, how good your quote is compared to the target price. It looks like this:



It can be used in Quotes, Price Lists, and many other places. To use it, simply have your calculation logic element return the object ResultGauge returned from calling `api.newGauge()`.

```
def g = api.newGauge()
g.setMin(0)
g.setMax(100)
g.setValue(20)
g.addSector(5,"#FF0000")
g.addSector(35,"#FFFF00")
g.addSector(null,"#00FF00")

quoteProcessor.addOrUpdateOutput(
  "ROOT",
  ["resultName": "Gauge",
   "resultLabel": "Gaugeeee",
   "resultType": "GAUGE",
   "result" : g ]
)
```



Edit Quote

Header Info

Name : P-12
Description : New Quote
Status : Draft
Workflow Status :
Effective Date : 05/03/2016
Expiry Date : 05/03/2016
External Reference :
Attachments : manage
User Group (Edit) :
User Group (View Details) :

Input Parameters [New Quote]

Customer : 40100659 BENNETT & FOUNTAIN LTD

Edit Header Text

Overview

Folder	Product Code	In	Total Stan...	Ta	Ne	Q	In	Ne
New Quote								
1.5 6491X ST...	20145157	11	58.00 %	£4	£5	40	£4,	£2,
1.5 6491X ST...	20145145	11	58.00 %	£4	£1	30	£3,	£3,
1.5 6491X SO...	20144659	10	58.00 %	£4	£4	20	£2,	£8

Calculation Results

Label	Calculation Re...	Message
Order Size	M	
Order Value Mix	Mix_Low_Mediu	
Region	North England	
Group Tier	T1	
Branch Tier	T3	
Competition Intensity	5	
Customer Parent Group	ANEW GRP	
Customer Class (Channel)	2A	
Net Order Value	6030	
Invoice Order Value	10426.30623642l	
Total Cost	4130	
Cluster ID	T1_5_T3_M_Mix_England	
Target CM(%)	0.085294039615l	
CM(%)	0.3150912106	
Gaugeeee		

Note: If you add the gauge to Overview as a column, a numerical value will be displayed here, not a graphical gauge as in Calculation Results.

How to Add a Matrix to Quote Header Output Fields

To display a matrix in in the quote header, use the following code snippet:

How to Add a Quote Link to Dashboard

Sometimes it might be useful for users to be able to open a quote directly from the dashboard.

Links to quotes can be included in a result matrix. See an example of a dashboard logic element below, which demonstrates various format types in a result matrix, including a link to a quote:

Definition of `m.linkCell(value, targetPage, targetPageState)`:

- **value** – Object that is passed, should contain the quote ID.
- **targetPage** – The name of the page to link to:
  - priceShopPage – For redirecting to quotes.
  - dashboardPage – For dashboards.
- **targetPageState**

How to Debug a Price List

To debug a price list, you can use the debug formula – but this is not always possible (it does not have the full context of the PL row). Then you can try the following trick:

1. Create a fake element for the calculation logic.
    a. Return some constant value in it (or whatever else, it does not matter).
    b. Set it to "manual override".
2. Save the logic.
3. Create a price list.
    a. Add Products to be tested.
    b. Add the fake element to the output.
    c. Calculate.
4. Open the price list and now change the value of the fake element for every line. Every time you change the value of the fake element, the line gets re-calculated! Hooray! So you can run the logic again and again – without waiting for the price list to be scheduled.

How to Add Entry to Matrix PP

To add an entry to a Matrix Pricing Parameter:

```
def pp = api.findLookupTable("MarginBySegment")
def entry = [:]
entry.lookupTableId = pp.id
entry.key1 = segment.Mfr
entry.key2 = segment.PBG
entry.key3 = segment.PBU
entry.key4 = segment.PRC
entry.key5 = segment.ProductDesignEligible
entry.attribute1 = pvs.low
entry.attribute2 = pvs.target
entry.attribute3 = pvs.high
api.addOrUpdate("MLTV5", entry)
```

It is important here to have thelookupTableId in the values map.

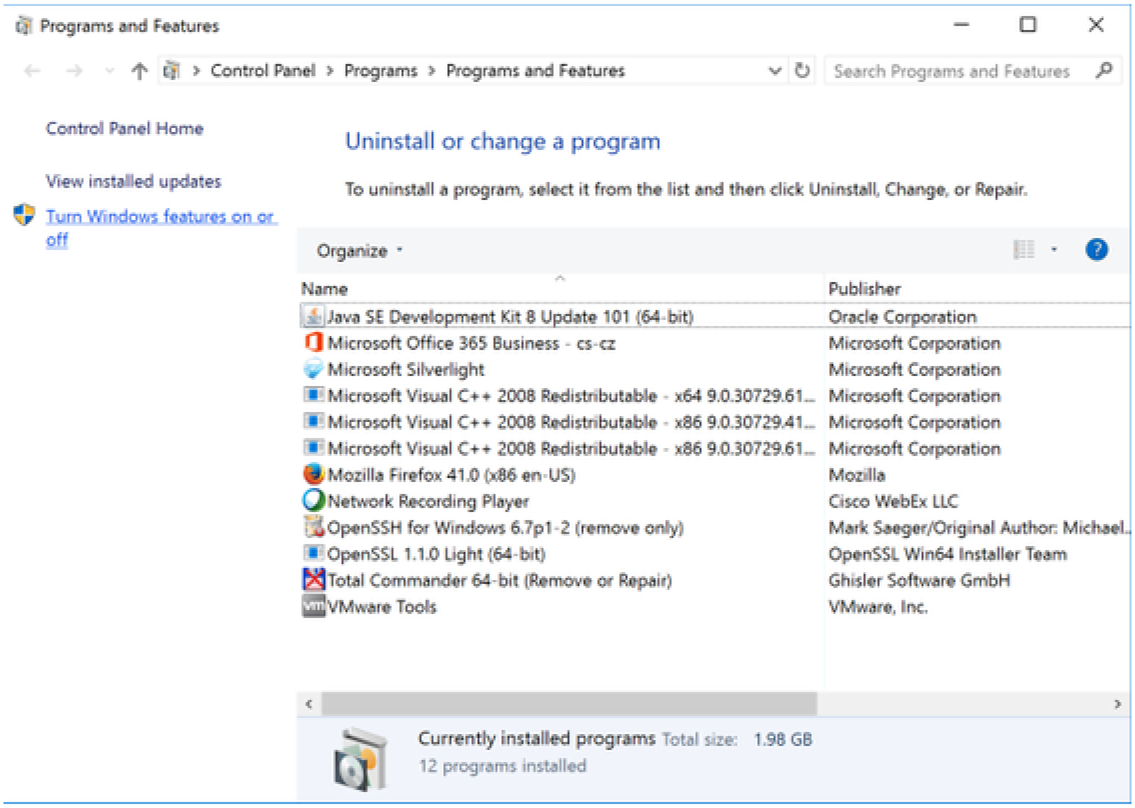
Alternatively, you can skip the findLookupTable call and specify the PP name directly in the values map:

```
def entry = [:]
entry.lookupTableName = "MarginBySegment"
entry.key1 = segment.Mfr
...
api.addOrUpdate("MLTV5", entry)
```

How to Install .NET 3.5 (Required for Excel Client) with Windows 10

Windows 10 (with the Anniversary update) has the .NET Framework already embedded. To have it recognized by the Pricefx Excel Client, you need to install .NET 3.5 using the Windows 10 steps:

1. Open the Control Panel.
2. Open Programs and Features.
3. In the left pane, click the 'Turn Windows features on or off' link.



Programs and Features
Control Panel > Programs > Programs and Features
Search Programs and Features
Control Panel Home
View installed updates
Turn Windows features on or off
Organize

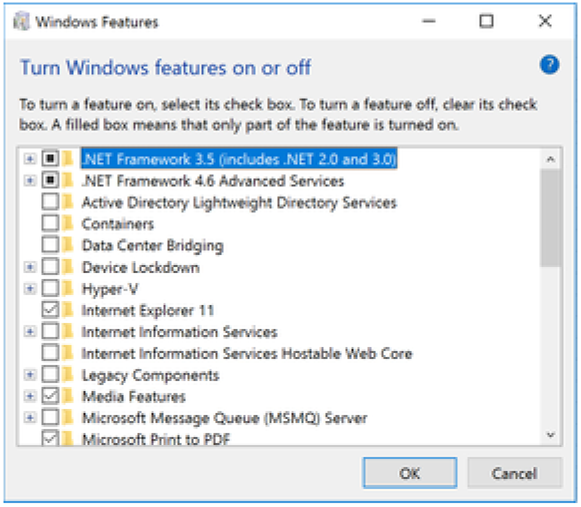
Uninstall or change a program

To uninstall a program, select it from the list and then click Uninstall, Change, or Repair.

Name	Publisher
Java SE Development Kit 8 Update 101 (64-bit)	Oracle Corporation
Microsoft Office 365 Business - cs-cz	Microsoft Corporation
Microsoft Silverlight	Microsoft Corporation
Microsoft Visual C++ 2008 Redistributable - x64 9.0.30729.61...	Microsoft Corporation
Microsoft Visual C++ 2008 Redistributable - x86 9.0.30729.41...	Microsoft Corporation
Microsoft Visual C++ 2008 Redistributable - x86 9.0.30729.61...	Microsoft Corporation
Mozilla Firefox 41.0 (x86 en-US)	Mozilla
Network Recording Player	Cisco WebEx LLC
OpenSSH for Windows 6.7p1-2 (remove only)	Mark Saeger/Original Author: Michael...
OpenSSL 1.1.0 Light (64-bit)	OpenSSL Win64 Installer Team
Total Commander 64-bit (Remove or Repair)	Ghisler Software GmbH
VMware Tools	VMware, Inc.

Currently installed programs Total size: 1.98 GB
12 programs installed

4. Enable the line called '.NET Framework 3.5 (includes .NET2.0 and 3.0)'.



Turn Windows features on or off

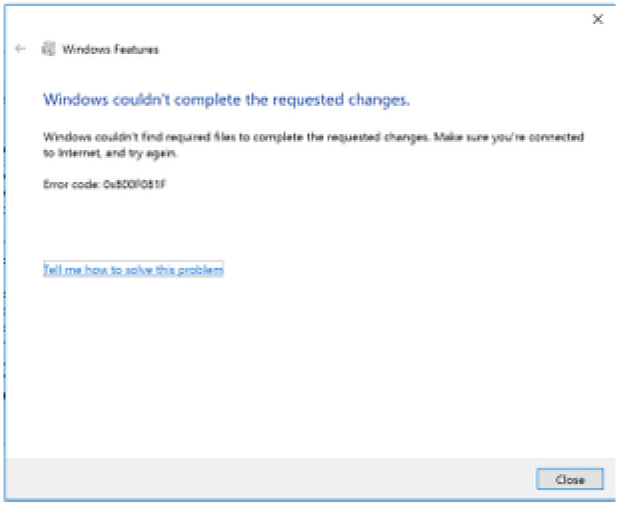
To turn a feature on, select its check box. To turn a feature off, clear its check box. A filled box means that only part of the feature is turned on.

- ■ .NET Framework 3.5 (includes .NET 2.0 and 3.0)
- ■ .NET Framework 4.6 Advanced Services
- □ Active Directory Lightweight Directory Services
- □ Containers
- □ Data Center Bridging
- □ Device Lockdown
- □ Hyper-V
- ☑ Internet Explorer 11
- □ Internet Information Services
- □ Internet Information Services Hostable Web Core
- □ Legacy Components
- ☑ Media Features
- □ Microsoft Message Queue (MSMQ) Server
- ☑ Microsoft Print to PDF

5. Click OK.
6. When asked, select 'Download files from Windows Update'.
7. Wait until the previous versions of .NET Framework are downloaded and installed.
8. If required, restart.

TROUBLESHOOTING

If you get this type of an error:



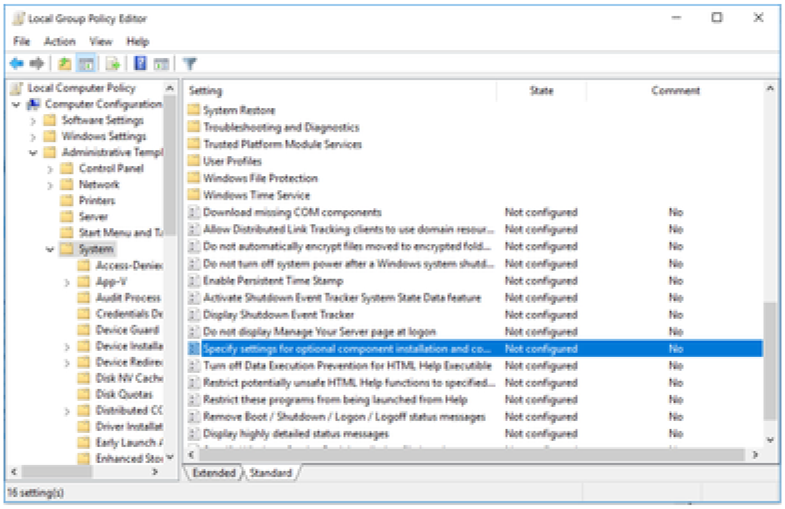
Windows couldn't complete the requested changes.

Windows couldn't find required files to complete the requested changes. Make sure you're connected to Internet, and try again.

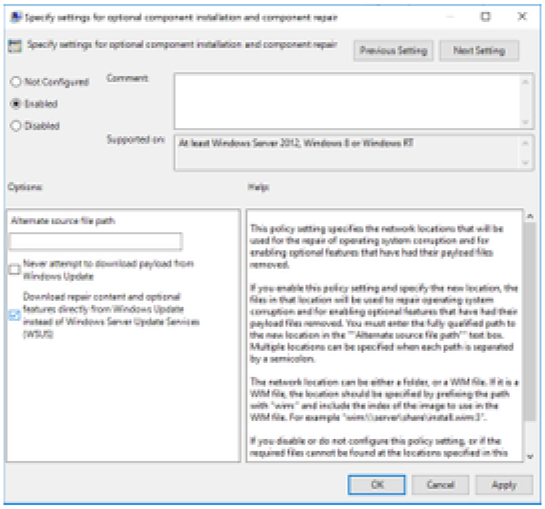
Error code: 0x800F081F

Tell me how to solve this problem

You can try to set the following:



Setting	State	Comment
System Restore		
Troubleshooting and Diagnostics		
Trusted Platform Module Services		
User Profiles		
Windows File Protection		
Windows Time Service		
Download missing COM components	Not configured	No
Allow Distributed Link Tracking clients to use domain resour...	Not configured	No
Do not automatically encrypt files moved to encrypted fold...	Not configured	No
Do not turn off system power after a Windows system shutd...	Not configured	No
Enable Persistent Time Stamp	Not configured	No
Activate Shutdown Event Tracker System State Data feature	Not configured	No
Display Shutdown Event Tracker	Not configured	No
Do not display Manage Your Server page at logon	Not configured	No
Specify settings for optional component installation and co...	Not configured	No
Turn off Data Execution Prevention for HTML Help Executible	Not configured	No
Restrict potentially unsafe HTML Help functions to specified...	Not configured	No
Restrict these programs from being launched from Help	Not configured	No
Remove Boot / Shutdown / Logon / Logoff status messages	Not configured	No
Display highly detailed status messages	Not configured	No


Specify settings for optional component installation and component repair
Previous Setting
Next Setting
Comment:
Options:
Help:
OK
Cancel
Apply

Specify settings for optional component installation and component repair

○ Not Configured
● Enabled
○ Disabled

Supported on: At least Windows Server 2012, Windows 8 or Windows RT

Alternate source file path

□ Never attempt to download payload from Windows Update

☑ Download repair content and optional features directly from Windows Update instead of Windows Server Update Services (WSUS)

This policy setting specifies the network locations that will be used for the repair of operating system corruption and for enabling optional features that have had their payload files removed.

If you enable this policy setting and specify the new location, the files in that location will be used to repair operating system corruption and for enabling optional features that have had their payload files removed. You must enter the fully qualified path to the new location in the ""Alternate source file path"" text box. Multiple locations can be specified when each path is separated by a semicolon.

The network location can be either a folder, or a WIM file. If it is a WIM file, the location should be specified by prefixing the path with "wim:" and include the index of the image to use in the WIM file. For example "wim:\\server\share\install.wim:3".

If you disable or do not configure this policy setting, or if the required files cannot be found at the locations specified in this

How to Run Excel Client on Mac

To run the Excel client on Mac using a virtual machine (VM):

1. Prepare the VM machine.
    a. Download a prepared VM from the Microsoft website: https://dev.windows.com/en-us/microsoft-edge/tools/vms/mac/
    b. Unzip it using Unarchiver, as suggested by Microsoft. (With another tool you may get incorrect results.) If you download the VM in a ZIP file for VMWare, you should have 3 files.
    c. Start Fusion.

    d. Import the existing VM: navigate to the *.OVF file, that is one of the 3 in the unzipped folder.
2. In the running Windows 10 instance install:
    a. MS .NET 3.5 (which includes 2.0 and 3.0 as well). This is required for the Excel client.
    b. MS Excel (You can use Office 2016, there is no need to register it.)

How to Find Set of Quotes Containing Given Products

1. Get all of the quotes (with any necessary filters):

```
qs = api.find("Q", i, step, null,
    Filter.equal("quoteStatus", quoteStatus),
    Filter.equal("workflowStatus", workflowStatus),
    Filter.equal("customerId", customerId))
```

2. Get the SKU or list of SKUs that you want to filter on.
3. Iterate through each line item in each quote. Within each iteration, loop through the list of SKUs and compare it to the quote line item's SKU:

```
def quotes = []
if (skus) {
  for (q in qs) {
    for (li in q.lineItems) {
      if (skus.contains(li.sku)) {
        quotes.add(q)
        break
      }
    }
  }
}
```

How to Generate Excel Reports from Calculation Logic

Goal: Generate custom MS Excel-based reports from a calculation logic

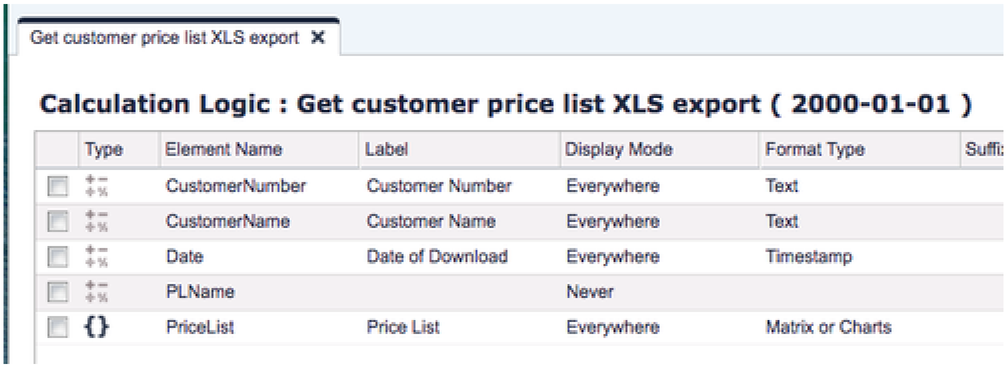
Use case example: Download a customer-specific price list from multiple sources (Price Lists, Price records etc)

Solution:

- Pricefx API method that executes the logic and creates an Excel output.
- Use the formulamanager.executeformula method and append the output type and the name of the template: output=xls&templateName=excel_export
  For more details, see the API documentation.
- Example: https://localhost/pricefx/demo/formulamanager.executeformula/Market-Pricing?output=xls&templateName=excel_export

GENERIC EXCEL OUTPUT

Logic example:

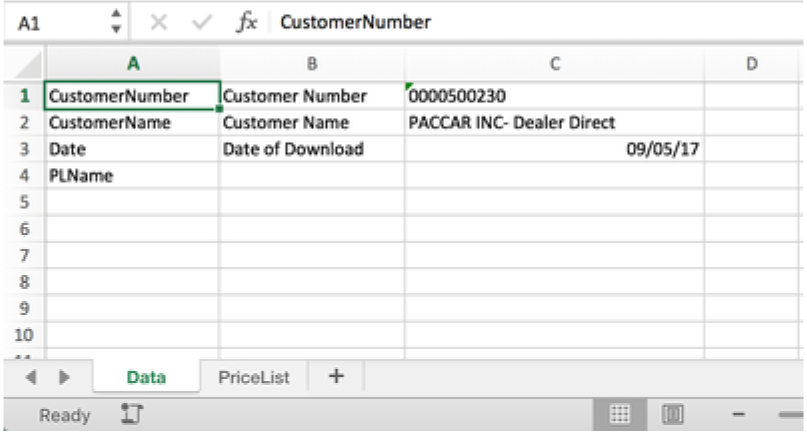


Get customer price list XLS export ✕

Calculation Logic : Get customer price list XLS export (2000-01-01)

	Type	Element Name	Label	Display Mode	Format Type	Suffi:
□		CustomerNumber	Customer Number	Everywhere	Text	
□		CustomerName	Customer Name	Everywhere	Text	
□		Date	Date of Download	Everywhere	Timestamp	
□		PLName		Never		
□	{}	PriceList	Price List	Everywhere	Matrix or Charts	

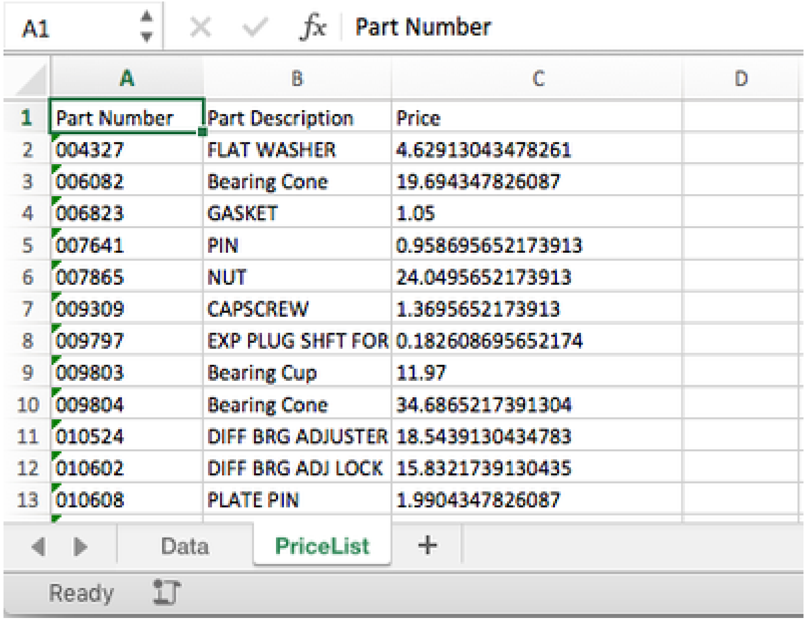
Corresponding Excel output example:



A1 | CustomerNumber

	A	B	C	D
1	CustomerNumber	Customer Number	0000500230	
2	CustomerName	Customer Name	PACCAR INC- Dealer Direct	
3	Date	Date of Download	09/05/17	
4	PLName			
5				
6				
7				
8				
9				
10				

Logic elements of the **matrix format type** will be rendered **in a separate sheet.** The tab's name reflects the matrix element name.



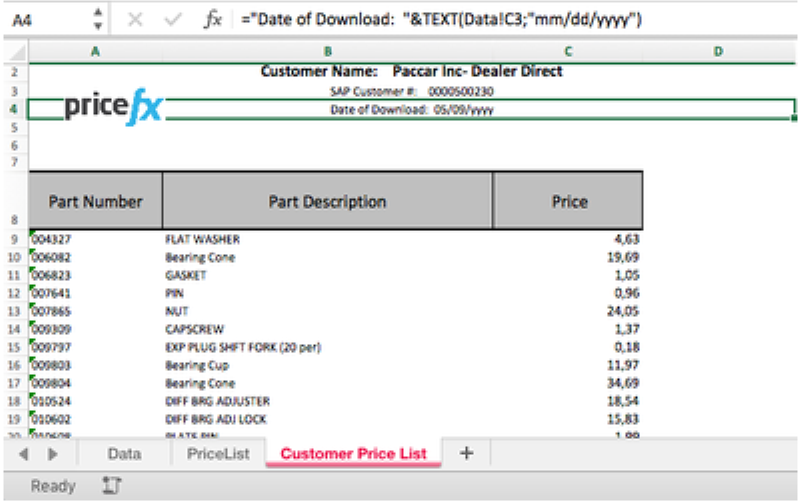
A1 | Part Number

	A	B	C	D
1	Part Number	Part Description	Price	
2	004327	FLAT WASHER	4.62913043478261	
3	006082	Bearing Cone	19.694347826087	
4	006823	GASKET	1.05	
5	007641	PIN	0.958695652173913	
6	007865	NUT	24.0495652173913	
7	009309	CAPSCREW	1.3695652173913	
8	009797	EXP PLUG SHFT FOR	0.182608695652174	
9	009803	Bearing Cup	11.97	
10	009804	Bearing Cone	34.6865217391304	
11	010524	DIFF BRG ADJUSTER	18.5439130434783	
12	010602	DIFF BRG ADJ LOCK	15.8321739130435	
13	010608	PLATE PIN	1.9904347826087	

EXCEL TEMPLATE

Similarly as for quotes and price lists, we can enhance generic output and save it as an Excel template. Such a template can be assigned to any PriceBuilder logic.

Please refer to Publishing Templates for more details.



A4 ="Date of Download: "&TEXT(Data!C3;"mm/dd/yyyy")

Customer Name: Paccar Inc- Dealer Direct

SAP Customer #: 0000500230

Date of Download: 05/09/yyyy

Part Number	Part Description	Price
004327	FLAT WASHER	4,63
006082	Bearing Cone	19,69
006823	GASKET	1,05
007641	PIN	0,96
007865	NUT	24,05
009309	CAPSCREW	1,37
009797	EXP PLUG SHFT FORK (20 per)	0,18
009803	Bearing Cup	11,97
009804	Bearing Cone	34,69
010524	DIFF BRG ADJUSTER	18,54
010602	DIFF BRG ADJ LOCK	15,83

DOWNLOAD LINK

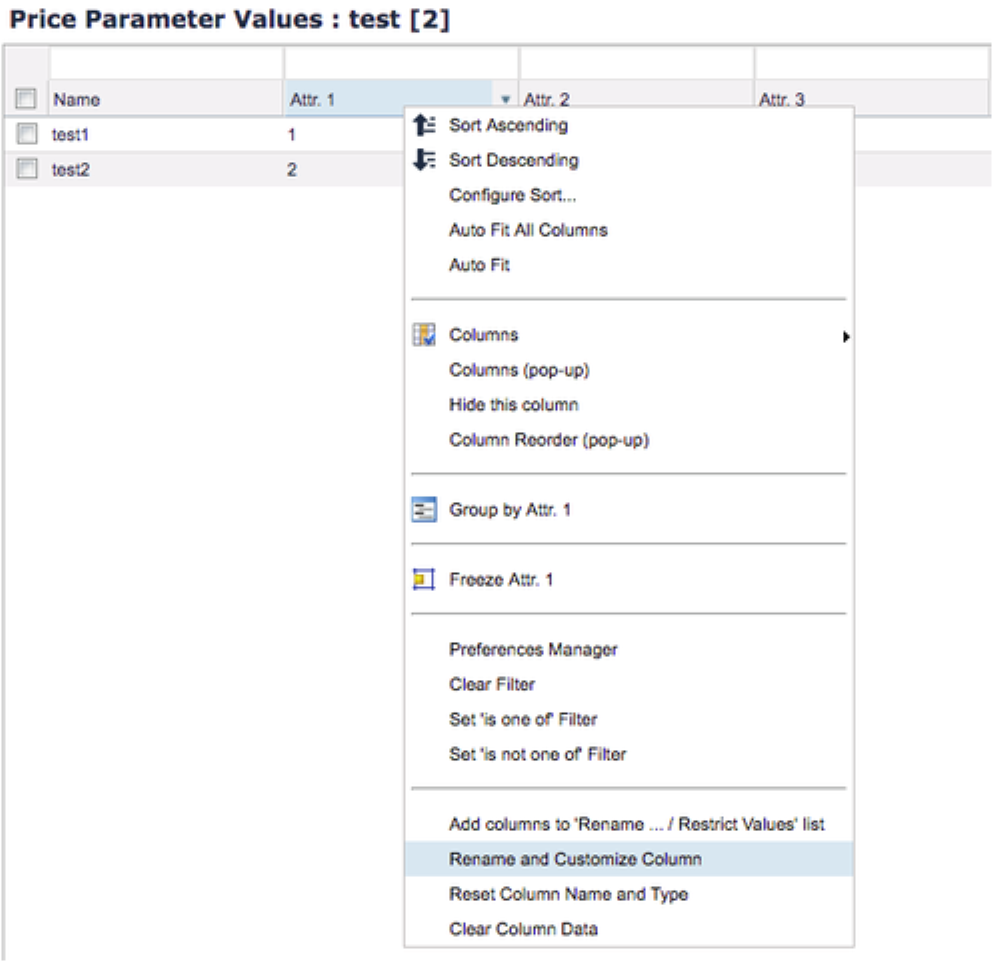
In case you would like to use a download link in the dashboard, please refer to How to Dynamically Generate Server Request from Dashboard.

How to Reference Values Between Two Price Parameter Columns

It is possible to create a reference between one Price Parameter which is taken as the source and another Price Parameter which takes values from the first one (available in that column as a drop-down menu). You can also reference other tables in the system, e.g., Customer, Product Extension.

This is useful if you need to restrict users to select values from another Price Parameter table or to set a new Price Parameter table quickly with data provided by the source Price Parameter.

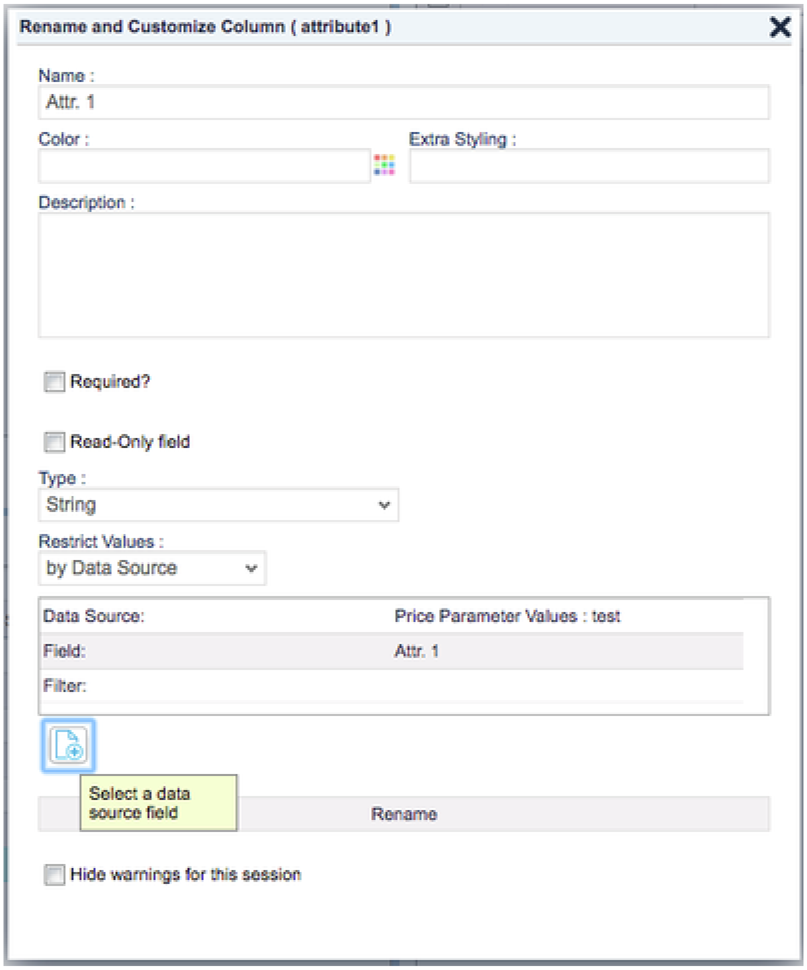
1. Go to Master Data > Price Parameters and select your source Price Parameter.
2. Right click the column header of the key or attribute from which you want make the reference and select 'Add columns to Rename... / Restrict Values list'. (For details see Customize Columns.)



Price Parameter Values : test [2]

	Name	Attr. 1	Attr. 2	Attr. 3
□	test1	1		
□	test2	2		

3. In the next pop-up window either keep the original label name or write your own.
4. After clicking OK, select the second Price Parameter which will take values from the first.
5. Right click the column header of the key or attribute you want make the reference to and click on 'Rename and Customize Column'. (For details see Customize Columns.)



Rename and Customize Column (attribute1)

Name : Attr. 1

Color :

Extra Styling :

Description :

□ Required?

□ Read-Only field

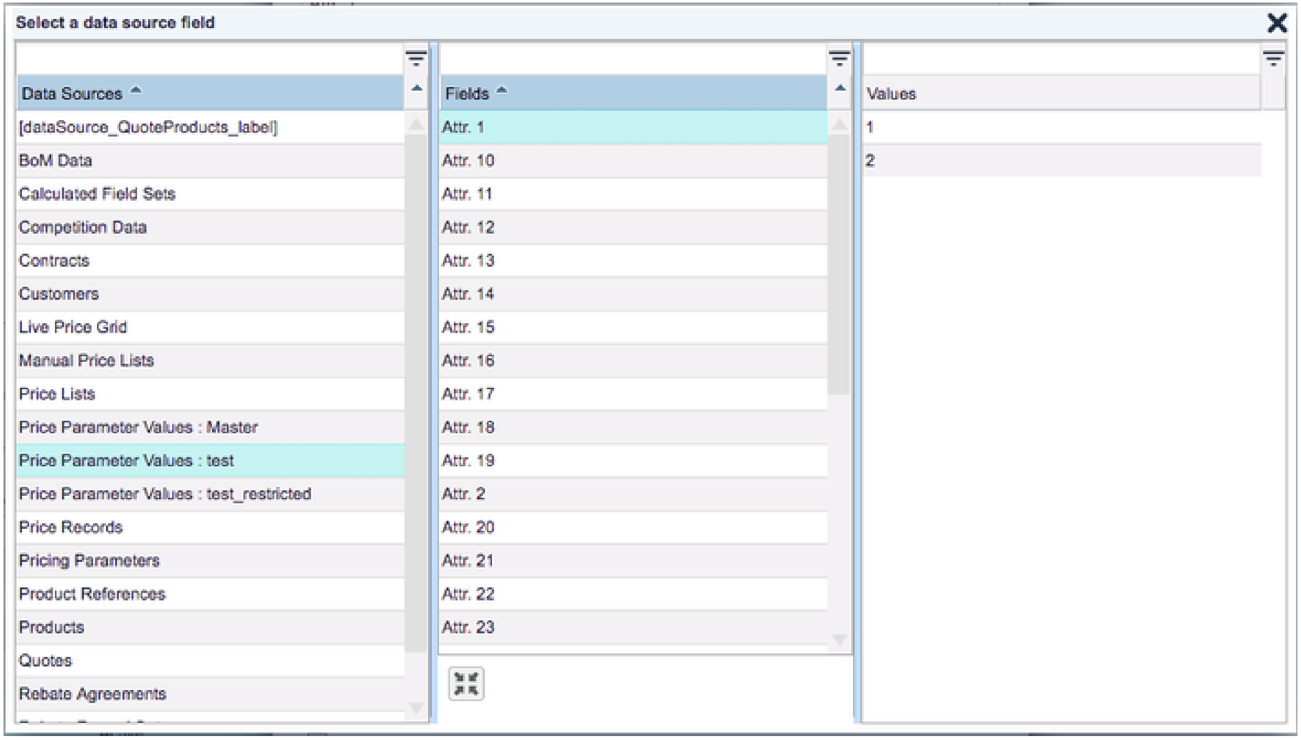
Type : String

Restrict Values : by Data Source

Data Source:	Price Parameter Values : test
Field:	Attr. 1
Filter:	

□ Hide warnings for this session

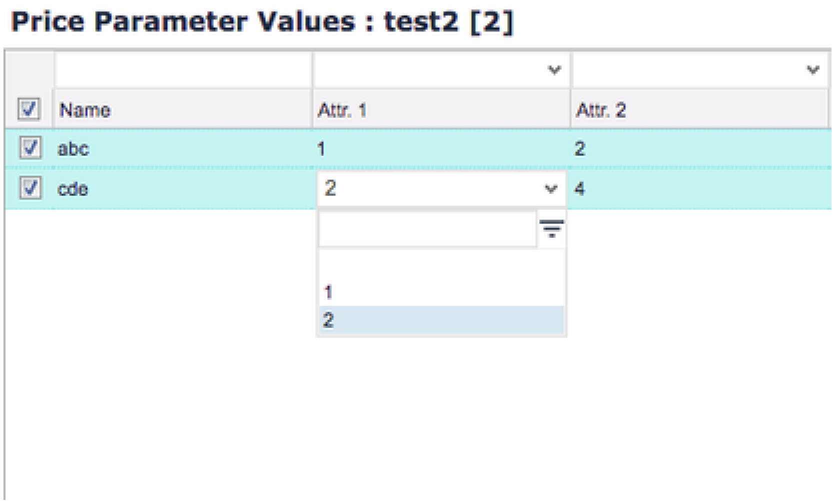
6. In the 'Restrict Values' field choose 'By Data Source' and click the Add button below.
7. Select your Data Source field that you named in Step 3, select the correct column in Fields and click the Select button.



Select a data source field

Data Sources	Fields	Values
[dataSource_QuoteProducts_label]	Attr. 1	1
BoM Data	Attr. 10	2
Calculated Field Sets	Attr. 11	
Competition Data	Attr. 12	
Contracts	Attr. 13	
Customers	Attr. 14	
Live Price Grid	Attr. 15	
Manual Price Lists	Attr. 16	
Price Lists	Attr. 17	
Price Parameter Values : Master	Attr. 18	
Price Parameter Values : test	Attr. 19	
Price Parameter Values : test_restricted	Attr. 2	
Price Records	Attr. 20	
Pricing Parameters	Attr. 21	
Product References	Attr. 22	
Products	Attr. 23	
Quotes		
Rebate Agreements		

8. Back in the Rename and Customize Column window click the Rename button.
9. Go to your desired column and double click the field – a drop-down menu with the restricted values should appear.



Price Parameter Values : test2 [2]

☑	Name	Attr. 1	Attr. 2
☑	abc	1	2
☑	cde	2	4

How to Populate Customer or Product Picker Created via API

If you do not use the standard Customer or Product picker but you create your own api.product() or api.customer() picker, you must explicitly retrieve the product or customer list – it is not done automatically.

The following example shows how to populate Customer input from a header logic:

```
cProcessor.addOrUpdateInput("ROOT", ["name": "CustomerGroup", "label":
"Customer(s)", "type": InputType.CUSTOMERGROUP, "value": [
"customerFieldLabel": "Unique Customer Id",
"customerFieldName" : "customerId",
"customerFieldValue": customerId
]])
```

How to Store Quote Inputs/Outputs in PriceAnalyzer

- PriceBuilder Logic
- PriceAnalyzer Logic
- Data Load

PriceBuilder Logic

Create a PB Feeder logic that emits the quote objects for use in the PA logic:

PriceAnalyzer Logic

Retrieve the target rowset from the Datamart:

Fetch the quote from the feeder logic:

Fetch header inputs/output values from the quote:

Fetch line-level inputs/output values from the quote (you can use api.getCalculableLineItemResult to fetch line-level outputs):

Create a map object mapping values to Datamart fields. (Map keys MUST match the Datamart field names). Add the row to the target rowset:

Create global variables for all field values:

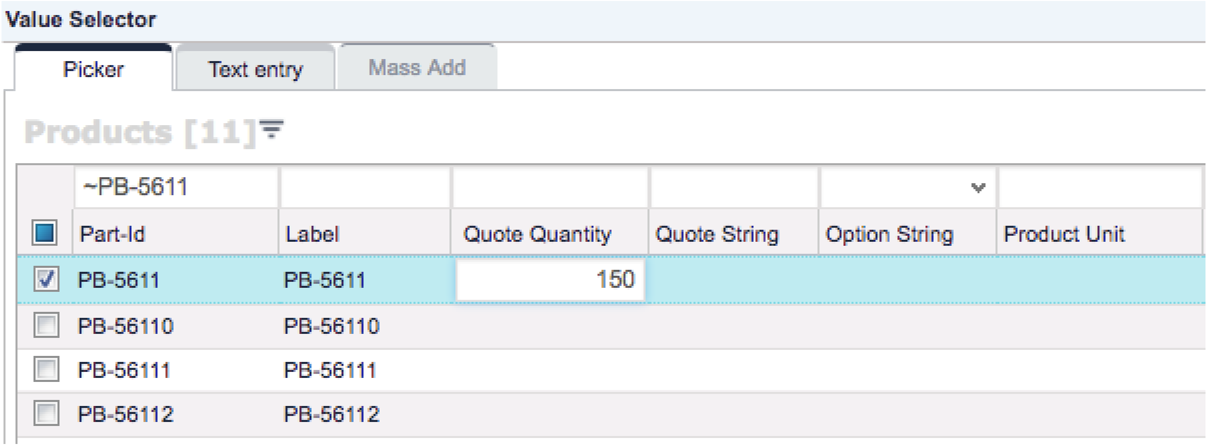
Create logic elements for each Datamart field. Return the global variables in each logic element with the Display Mode "Everywhere", so they can be accessed in the Data Load.

Data Load

If the target is a Datamart, create a calculation data load for Datamart that runs incrementally and assign the appropriate PA and PB logics as the Formula and Feeder logics respectively. If the target is a Data Source, update the existing 'Flush' type Data Load for the Data Source by configuring the Formula and Feeder logics in the Calculation section.

How to Add Inputs to Product Selector

Inputs, which are displayed in the quote's Input Parameters section can be made available to users in the product selector (as columns), so that users can select products and enter inputs in one step.



☐	Part-Id	Label	Quote Quantity	Quote String	Option String	Product Unit
	~PB-5611					
☑	PB-5611	PB-5611	150			
☐	PB-56110	PB-56110				
☐	PB-56111	PB-56111				
☐	PB-56112	PB-56112				

How to set it up?

Use the application property 'quoteItemSelectorInputs' to define the list of inputs. The property is set in the Advanced Configuration Options section.

The value of the property will look similar to this:

The inputs defined by the application property will always be shown in the Product selector. Inputs that do not match any inputs present in the current calculation logic will be ignored.

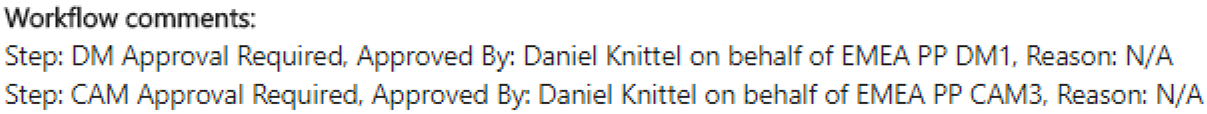
Only the USERENTRY, STRINGUSERENTRY and OPTION input types are supported.

How to Add Approver to Workflow Email Template

To access the approver of the current workflow step, use:

If you want to have a list of all approvers of a quote, you can iterate through the workflow history steps with the code below:

The above code will look like this in a workflow email:

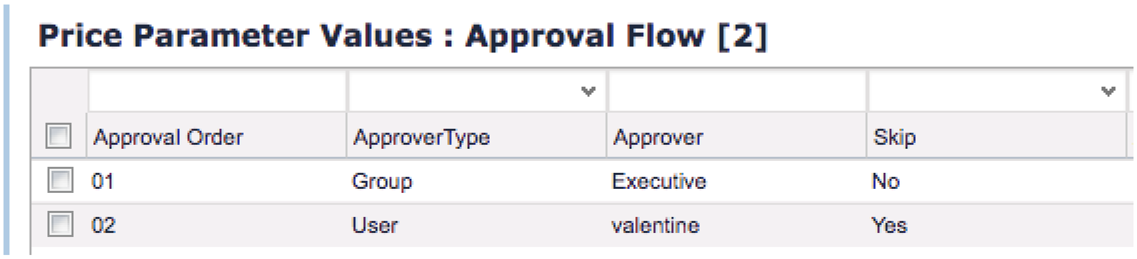


Workflow comments:
Step: DM Approval Required, Approved By: Daniel Knittel on behalf of EMEA PP DM1, Reason: N/A
Step: CAM Approval Required, Approved By: Daniel Knittel on behalf of EMEA PP CAM3, Reason: N/A

How to Create Parametric Approval Workflow

If a customer requires to be able to add or remove an approval step at any point without touching the code, this can be achieved by creating a pricing parameter table with the following columns:

- **Approval Order** – Defines the order in which the approval steps are sequenced.
- **Approver Type** – Defines whether approver is a group or an individual user.
- **Approver** – Defines either group name (for the Group Approver type) or a user ID (for the User type).
- **Skip** – Allows to remove an approver without deleting the record from the table (which later allows to quickly add the approver back to the workflow).



Price Parameter Values : Approval Flow [2]

□	Approval Order	ApproverType	Approver	Skip
□	01	Group	Executive	No
□	02	User	valentine	Yes

Keep in mind that the Approval Order must be a string, and as such it is recommended to pad single digits with a 0 - to make sure that if there are more than 9 approvers they are sorted correctly. In the unlikely event that there are more than 99 approvers another 0 must be added before the number.

In the Configuration > Workflows > Workflow Logics create the following workflow logic:

```
if (quote?.lineItems == null) return

def ppTable = api.findLookupTable("ApprovalFlow")
def approvers = api.find("MLTV", 0, "name",
Filter.equal("lookupTable.uniqueName", "ApprovalFlow"))

for (li in quote?.lineItems){
  if (li.folder) continue
  for (e in li?.outputs)
  {
      for (approver in approvers) {
          if (approver.attribute1 == "Group" && approver.attribute3 !=
"Yes") {
           workflow.addApprovalStep("Approver" +
approver).withUserGroupApprover(approver.attribute2).setReason("Requires
" + approver.attribute2 + " approval")

          }else if (approver.attribute1 == "User" && approver.attribute3
!= "Yes") {
            workflow.addApprovalStep("Approver" +
approver).withApprover(approver.attribute2).setReason("Requires " +
approver.attribute2 + " approval")
          }
        }
      break
   }
}
```

The full code is available in Git - FloorPriceApprovalSim.groovy

How to Create Button in Email Template

You can use CSS to style an href into a button in an email template using the following code as an example:

This will result in this button in the workflow email:

How to Use Secondary Key to Create Matrix Price Lists

Written by:	Valentine Azbelle

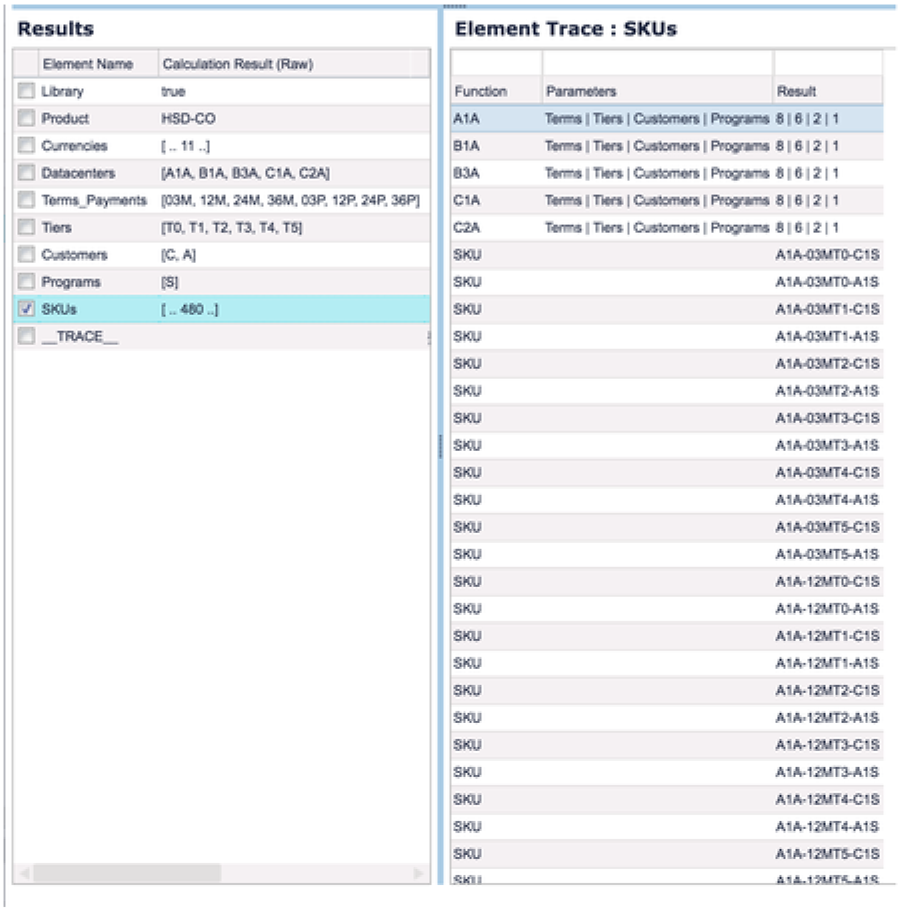
In order to create a matrix price list, two logics are required – one generates the key, the other calculates the price list based on that key. The key generator logic must be specified as Matrix logic in the Price List setup.

Define the matrix key generator (matrix pricing logic):

1. Go to Configuration > Calculation Logic > Generic Logic. Create a new logic and set Nature to Matrix.
2. Define the logic's elements. Create one visible element which will return a list of keys generated based on input parameters. Below is an example of keys generated from Datacenters, Terms_Payments, Tiers, Customers, and Programs elements.

```
def datacenters = api.getElement("Datacenters")

def skus = []

// data centers
for (dc in datacenters) {
  def datacenter = dc.name
  def dcCode = dc.attribute2.replaceAll(" ", "_")

  def terms_payments = api.getElement("Terms_Payments")
  def tiers = api.getElement("Tiers")
  def customers = api.getElement("Customers")
  def programs = api.getElement("Programs")

  // terms
  for (tr in terms_payments) {

    // tiers
    for (ti in tiers) {

      // customers
      for (c in customers) {

        // programs
        for (pr in programs) {
          def sku = datacenter + "-" + tr + ti + "-" + c + "1" + pr
          skus.add(sku)
        }
      }
    }
  }
}

for (sku in skus) {
  api.trace("SKU", null, sku)
}

return skus
```

The code returns a list of SKU’s which are being used as secondary keys for Matrix Price List calculation. A sample trace of the above logic looks like this:



Results

	Element Name	Calculation Result (Raw)
□	Library	true
□	Product	HSD-CO
□	Currencies	[.. 11 ..]
□	Datacenters	[A1A, B1A, B3A, C1A, C2A]
□	Terms_Payments	[03M, 12M, 24M, 36M, 03P, 12P, 24P, 36P]
□	Tiers	[T0, T1, T2, T3, T4, T5]
□	Customers	[C, A]
□	Programs	[S]
☑	SKUs	[.. 480 ..]
□	__TRACE__	

Element Trace : SKUs

Function	Parameters	Result
A1A	Terms | Tiers | Customers | Programs	8 | 6 | 2 | 1
B1A	Terms | Tiers | Customers | Programs	8 | 6 | 2 | 1
B3A	Terms | Tiers | Customers | Programs	8 | 6 | 2 | 1
C1A	Terms | Tiers | Customers | Programs	8 | 6 | 2 | 1
C2A	Terms | Tiers | Customers | Programs	8 | 6 | 2 | 1
SKU		A1A-03MT0-C1S
SKU		A1A-03MT0-A1S
SKU		A1A-03MT1-C1S
SKU		A1A-03MT1-A1S
SKU		A1A-03MT2-C1S
SKU		A1A-03MT2-A1S
SKU		A1A-03MT3-C1S
SKU		A1A-03MT3-A1S
SKU		A1A-03MT4-C1S
SKU		A1A-03MT4-A1S
SKU		A1A-03MT5-C1S
SKU		A1A-03MT5-A1S
SKU		A1A-12MT0-C1S
SKU		A1A-12MT0-A1S
SKU		A1A-12MT1-C1S
SKU		A1A-12MT1-A1S
SKU		A1A-12MT2-C1S
SKU		A1A-12MT2-A1S
SKU		A1A-12MT3-C1S
SKU		A1A-12MT3-A1S
SKU		A1A-12MT4-C1S
SKU		A1A-12MT4-A1S
SKU		A1A-12MT5-C1S
SKU		A1A-12MT5-A1S

Define a price list logic which will reference the matrix logic and retrieve the key list:

1. Go to Configuration > Calculation Logic > Generic Logic. Create a new logic.
2. Create an element which retrieves the secondary key using the method `api.getSecondaryKey()`.
3. The returned key can be split into various components which will be used to calculate the price based on their variations.

```
def key = api.getElement("Debug")=="No" ? api.getSecondaryKey() :
"B1A-36PT5-C1S"

def parts = key?.tokenize("-")

return [
  dc: parts[0],
  term: parts[1][0..1],
  payment: parts[1][2],
  tier: parts[1][3..4],
  program: parts[2][0],
  version: parts[2][1],
  priceType: parts[2][2]
]
```

4. This makes it possible to extract a secondary key component for subsequent substitutions. Like this:

```
return api.getElement("SecondaryKey")?.dc  //for Datacenter
```

or

```
return api.getElement("SecondaryKey")?.term  //for Term
```

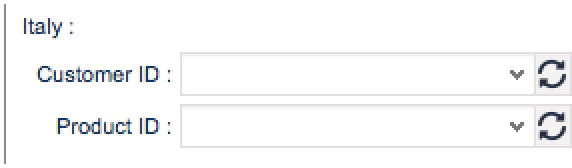
5. Subsequent logic performs substitutions to build the SKU, look up currency, and calculate prices for the price list.

How to Create Cascading Filters

You can define cascading filters for dashboards, where the selection made in one filter determines the options available in the dependent filter.

The following code sample displays to the end user a clickable 'Generic filter' link, which opens a FilterBuilder window.

The following sample shows 2 dim filters (without default values) to the end user.
The (optional) underlying generic filter (country == Italy) is hidden and not modifiable by the user.
Note that the dim filter fields are in a sub-section with the user entry's name as the title.

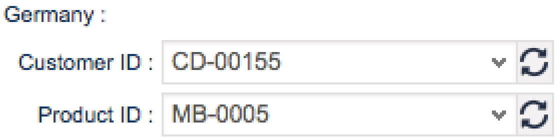


Italy :

Customer ID :

Product ID :

The following sample shows two dim filters with default values (CD-00155, MB-0005) to the end user.
The (optional) underlying generic filter (country == Germany) is hidden and not modifiable by the user (do not confuse with the section title, which is taken from the user entry name).



Germany :

Customer ID : CD-00155

Product ID : MB-0005

How to Create Portlet Alternating Between Chart and HTML

This functionality is available as of the Cosmopolitan release.

The following example shows a dashboard logic which can switch to an HTML (e.g. such a nicely looking error message; here represented by the PFX logo) based on a condition (here represented by the 'decision').

```
def fields = [
  [name: "String value", type: FieldFormatType.TEXT],
  [name: "Integer value", type: FieldFormatType.INTEGER],
  [name: "Numeric value", type: FieldFormatType.NUMERIC],
  [name: "Percent value", type: FieldFormatType.PERCENT],
  [name: "Date value", type: FieldFormatType.DATE],
];


def m = api.newMatrix(fields.collect{it -> it.name});
def rows = [];
def d = new Date();

for (int i=0; i< 500; i++) {
  rows << ["string text "+i, i, 0.11*i, 0.01*i, d-i];
}


fields.each {field ->
  m.setColumnFormat(field.name, field.type);
}

rows.each {row ->
  m.addRow((0..row.size()-1).collectEntries{index ->
    [fields[index].name, row[index]]
  });
}

///end matrix

def controller = api.newController()

controller.addHTML("<img
src='https://www.pricefx.eu/site/wp-content/uploads/2016/02/logo.png'
width=147 height=36/>")


return decision() ? m : controller;

boolean decision() {
  def m2 = new Date()
  return new BigDecimal(m2.getTime()).remainder( 2 ) != 0
}
```

How to Create Quote Line Items from Header Logic

Sometimes it is required to pass values from a header logic to line items that this logic creates. You can use the HIDDEN input type and reference it in the calculation logic.

The following code sample creates a new quote line with parameters fetched from the header logic:

In the line calculation logic, this value then can be retrieved by:

Notes:

- This way you can also set default values of "real" input parameters as determined by the line calculation logic. Set the value in the header logic and make sure that the names of the parameter match in both logics.
- If you use this method only to transport values between header & line, set the parameter type to HIDDEN. The value can also be a map to transport more structured data.
- Any parameter that is non-HIDDEN and not determined by the line logic will be removed.
- Be aware that this hidden data is also exposed to the user (it is part of the quote response). It is not visible in the UI, but it is neither a secure storage! Also, try to keep the data volume reasonably low.

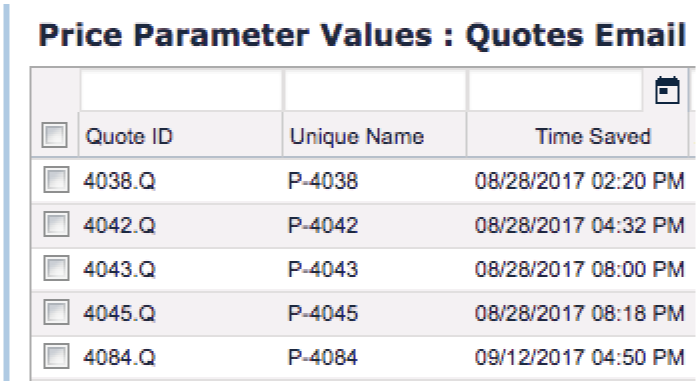
You can also set default values for items created using the quoteStructure class:

How To Generate Approval Required E-mail On Quote Save

If a client's workflow requires that the quote remains editable after being submitted for approval and notifications must be sent to the approvers, this can be achieved by generating the notification e-mails when the quote is saved (rather than submitted). The basic concept entails creating a “Submit To” flag with “Yes”/”No”/null options and a price parameter table to keep track of the generated e-mails (to prevent the e-mails from being generated more than once).

- When the quote is saved with “Submit To” == null -> nothing happens.
- When the quote is saved with “Submit To” == “Yes” -> a notification e-mail is generated to the appropriate parties and the quote gets entered into a tracking table to avoid duplicate e-mail generation.
- When the quote is saved with “Submit To” == “No” -> the quote record is deleted from the tracking table so that when the quote is saved with “Submit To” == “Yes” the e-mail is generated again.

Create a Price Parameter MLTV table (in this case called “QuotesEmailSent”) as follows:



Price Parameter Values : Quotes Email

□	Quote ID	Unique Name	Time Saved
□	4038.Q	P-4038	08/28/2017 02:20 PM
□	4042.Q	P-4042	08/28/2017 04:32 PM
□	4043.Q	P-4043	08/28/2017 08:00 PM
□	4045.Q	P-4045	08/28/2017 08:18 PM
□	4084.Q	P-4084	09/12/2017 04:50 PM

Create a Calculation Flow Logic with the following code:

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


def quotesSaved = true
int i = 0
int step = 200
def qs = []
def quotes = []
def table = api.findLookupTable("QuotesEmailSent")
def tableValues = api.findLookupTableValues("QuotesEmailSent")
def quoteList = []
//create a list of quotes in the PP tracking table
for (quote in tableValues) {
  quoteList.add(quote.name)
}
```

```
//find all users in the approver group

def ug = api.find("U", Filter.equal("groups.uniqueName", "Marketing
Manager"))

//create the list of e-mails of the users in the approver group

def emailList = []

for (e in ug) {
  emailList.add(e.email)
}

//iterate over saved quotes

while (quotesSaved) {
  qs = api.find("Q", i, step, null)

  if (qs.size() != 0) {
    quotes.addAll(qs)
    i += step
  } else {
    quotesSaved = false
  }

/*Find the quotes with the “Submit To” flag switched to “Yes” that are
not in the tracking table,
and generate the e-mails adding the quote(s) to the tracking tables.
Find the quotes with the “Submit To” flag switched to “No” and remove
the quote(s) from the tracking table*/

  for (q in qs) {
    def clic = api.getCalculableLineItemCollection(q?.typedId)
    for (lineItem in clic?.lineItems) {
        if (lineItem.folder) continue
    }
 for (input in clic?.inputs) {

    if (input.name == "submitForApproval" && input.value == "Yes" &&
!quoteList.contains(q.typedId)) {
          for (email in emailList) {
            api.sendEmail(email, "Submitted For Quote",
"https://www.pricefx.eu/priceFxWeb.html?locale=en_US&targetPageState=" +
q.typedId + "&targetPage=priceShopPage#priceShopPage")
          }

           updateRecord = [lookupTableId : table.id, "name" : q.typedId,
"attribute1" : q.uniqueName, "attribute2" : q.lastUpdateDate]
           api.addOrUpdate("MLTV", updateRecord)
        } else if (input.name == "submitForApproval" && input.value ==
"No") {
          existingRecord = [lookupTableId : table.id, "name" :
q.typedId, "attribute1" : q.uniqueName, "attribute2" : q.lastUpdateDate]
          api.delete("MLTV", existingRecord)
         }
```

```
      }
   }
  }
```

Create a calculation flow which will run this logic every 5 minutes (or as often as you and the client deem reasonable).

In case there are more than one approval levels you will need to create a "Submit To" flag and a PP tracking table for each level.

The full code (including HTML formatting for the e-mail) is avaliable in Git - Copy_of_GenerateEmailOnQuoteSave_VA1.groovy

How To Disable Submit Quote Button in QuoteConfigurator

For new options how to disable the Submit button, see How to Hide, Disable or Make Required Fields/Buttons.

How to Hide, Disable or Make Required Fields/Buttons

Since the Cuba Libre version, you can:

- Manage visibility and editability of quote/contract/agreement header fields.
- Hide the Submit and Save buttons.

Fields

To modify the visibility or editability of header fields, use the `qp.clearRenderInfo("fieldname") or qp.setRenderInfo("fieldname" ,"key","value")` methods.

Header Logic Snippet

```
def qp = quoteProcessor
if (qp.isPrePhase()){
 qp.setRenderInfo("userGroupEdit","hide",true)
 }
```

The `qp.setRenderInfo` method has three arguments:

- Field name – All quote/contract/agreement fields shown in the header input portlet are currently supported. The field names correspond to the names in the JSON data structure.
- Key – Header fields support these three keys: `hide`, `required`, `disabled`.
- Value – Value attribute of the boolean type

Running the code above generates a JSON sub-structure on the quote header looking like this:

```
"renderInfo" : {
  "userGroupEdit" : {
  "hide" : true
  }
 },
```

The UI can now evaluate the instructions as it sees fit, i.e. the field names as well as the available key value pairs are not restricted/enforced by the backend.

Buttons

You can hide a button using the `qp.setRenderInfo("buttonname","key","value")` method.

- Button name – Currently only two values are suported: `saveButton` and `submitButton`.
- Key – Only one key is supported: `hide`.

- Value – Value attribute of the boolean type

Header Logic Snippet

```
def qp = quoteProcessor
if (qp.isPrePhase()){
 qp.setRenderInfo("submitButton","hide",true)
 }
```

How to Update Object References Regularly

As customers and attributes may change any time, it is handy to run an automatic background job to update the object references regularly on a particular partition.

A typical use case: We have contracts related to a customer group. There is a new customer added to that group. When displaying contracts of that newly added customer, we want to make sure the data is up-to-date and includes all of the contracts of the corresponding customer group.

To ensure a timely object references update, we need to trigger `addObjRefAction`. This can be done using a Calculation Flow which is set to a reasonable frequency (e.g. once a day).

```
actionBuilder.addObjRefAction().setFilter(filter).setPartition(partition).setTypeCode('typeCode')
```

For the above mentioned case with contracts, we would adjust "typeCode" to be "CT", "partition" to be the partition you need it for and filter as needed.

How to Set Up a Buy-One-Get-One-Free Promotion

Business use case:

You want to create a promotional campaign, in which your customers, when they buy a specified product, receive a free bonus product. In practice, this means that after adding a certain product to a quote, another product is added automatically with a zero price tag.

Requirements:

- After choosing a customer and adding product X to the quote, product Y should be added to the quote automatically.
- After changing the quantity of product X and clicking 'Recalculate Changes' or 'Recalculate', the quantity of product Y should change accordingly and the user input should be read-only.
- The 'Requested Price' input should be set to 0 and read-only for product Y.
- After deleting a quote line containing product X and clicking 'Recalculate Changes' or 'Recalculate', the line with product Y should be deleted.

Solution:

The solution will consist of two parts. First, we define the promotion in PromotionManger and in the second part, we will handle the promotion in the quote.

PromotionManager

First of all we need to create a logic for the promotion. The logic will enable us, when creating a new contract, to determine which products will be promoted, to which customer groups it will apply and set the priority in case that we create more promotions.

Its conditions will be stored in Price Records, and for this reason, the PR column names must be the same as the names of the elements in the promotion calculation logic. We need to store the following information: ContractTermType, PromotionalProduct, PromotionPriority (change the names of PR columns accordingly). In this example we use the following names: PromotionPriority-attribute1, ContractTermType-attribute5, PromotionalProduct-attribute6.

In Configuration > Calculation Logic > Contracts, we create the promotion logic, which will consist of five elements.

The first element creates a user input where the promoted product is selected:

The second element creates a user input for selecting the free promotional product:

The third element creates a user input for Customer Group:

The fourth element creates a user input for assigning a priority level to the promotion:

And finally the fifth element saves the information about ContractTermType to the Price Records:

When we have all the information about our promotion safely stored in Price Records, we can use it in quotes.

QuoteConfigurator

To achieve the functionality described at the beggining of this article, we need to transport information from the quote header logic (that adds the bonus item) to the line logic. We will use HIDDEN type inputs to do that (see also How to Create Quote Line Items from Header Logic).

First, we will create a quote header logic:

Then we need to create a logic for the quote line items in Configuration > Calculation Logic > Generic Logic. This logic will have three elements.

The first element creates a user input, where the user indicates the desired quantity of the product:

The second element creates a user input, where the user indicates the requested price:

In the third element, we just want you to show, that in the line logic we can read a value of the input which has been added in Header Logic.

Everytime you add product X to your quote, the header and line logics will be executed in the following order:

1. Header logic in the pre-phase.
2. Line logic for new product (X).
3. Line logic for the new, added by the header logic, product (Y).
4. Header logic in the post-phase.

For your convenience, all the code samples are downloadable below:

XforYHeader.7z

XforYQuoteLineItem.7z

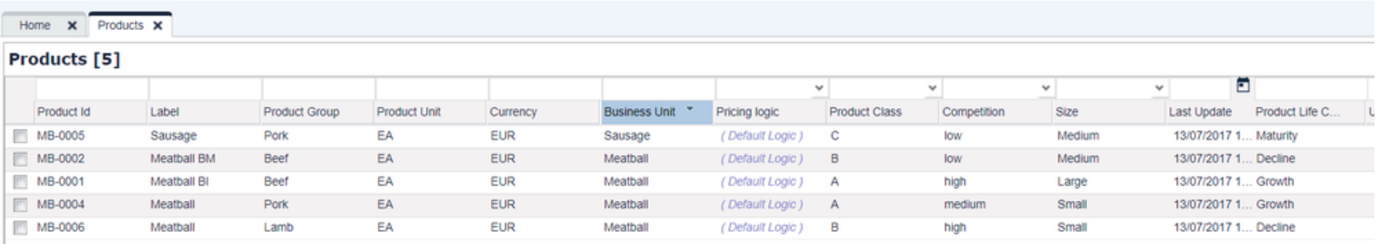
XforYContractLogic.7z

How to Automatically Organize Products into Folders

In QuoteConfigurator, folders can be logically created based on product attributes.

Each time a user adds products, the code will run and automatically place the items in the folders.

In this example, the folders are based on Product Class.



Products [5]

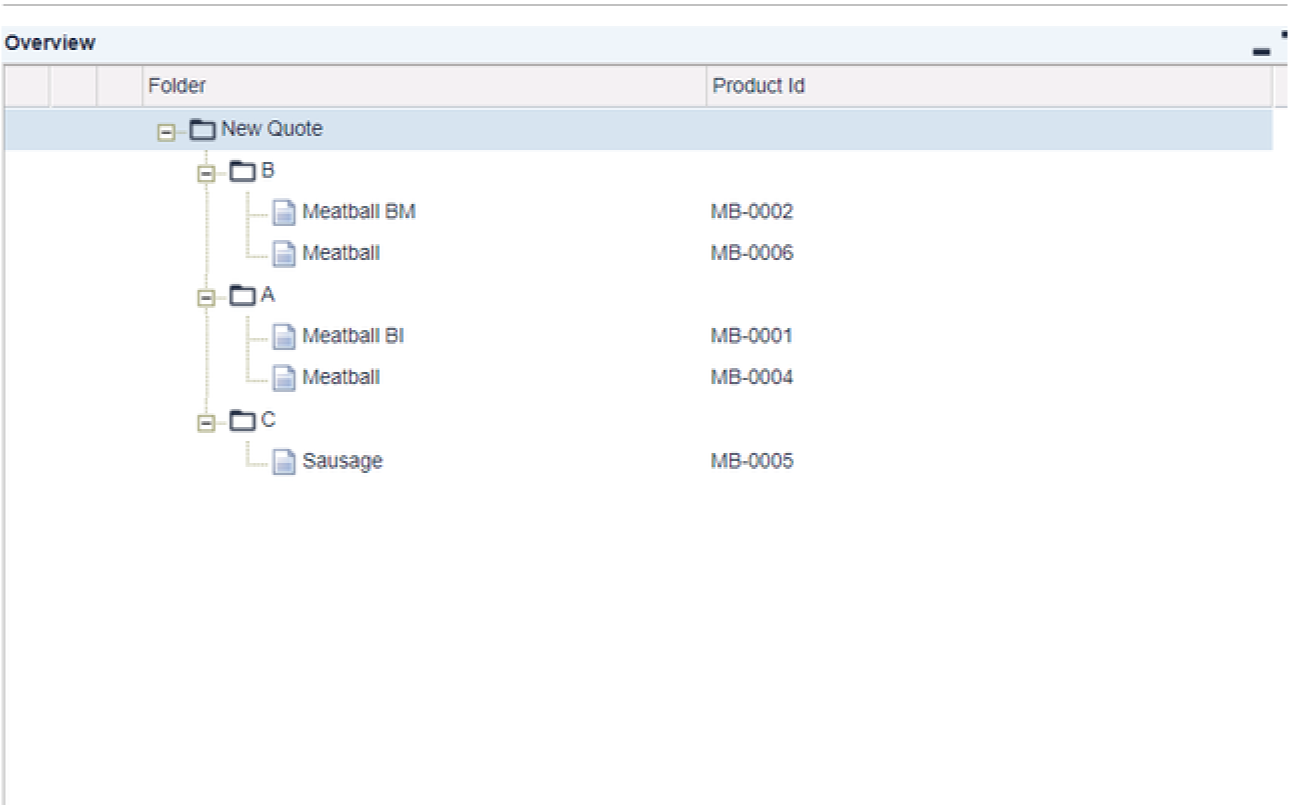
	Product Id	Label	Product Group	Product Unit	Currency	Business Unit	Pricing logic	Product Class	Competition	Size	Last Update	Product Life C...
□	MB-0005	Sausage	Pork	EA	EUR	Sausage	*(Default Logic)*	C	low	Medium	13/07/2017 1...	Maturity
□	MB-0002	Meatball BM	Beef	EA	EUR	Meatball	*(Default Logic)*	B	low	Medium	13/07/2017 1...	Decline
□	MB-0001	Meatball BI	Beef	EA	EUR	Meatball	*(Default Logic)*	A	high	Large	13/07/2017 1...	Growth
□	MB-0004	Meatball	Pork	EA	EUR	Meatball	*(Default Logic)*	A	medium	Small	13/07/2017 1...	Growth
□	MB-0006	Meatball	Lamb	EA	EUR	Meatball	*(Default Logic)*	B	high	Small	13/07/2017 1...	Decline

The following code will be in the QuoteHeader logic:

1. Ensure that the code only runs in the Pre Phase. Items cannot be deleted or moved in the Post Phase.
2. Find items in the quote that are products and are not already present in the folders. We do not want to touch the items that are already sorted.
3. Search for the Product Class of each item that meets the above criteria.
4. Go in cycles through the line items and delete each of them. They will be re-added later.
5. Create a new QuoteStructure to add to the quote later.
6. Determine which folders to create. Some Product Classes may already have folders and we do not want duplicates.
7. Find the added skus associated with the particular Product Class.
8. If the folder needs to be created, add the items to the folder's QuoteStucture which you have created earlier. Be sure to use createPart, as addPart will cause the code to run again and result in duplicate items.
9. If the folder already exists, find the folder's line id, and add the items to the folder. Alternatively, you could detect this condition earlier and use moveItem instead of deleting.

Finally, add the QuoteStructure to the quoteProcessor item.

The end result will look like this:



Overview

Folder	Product Id
New Quote	
B	
Meatball BM	MB-0002
Meatball	MB-0006
A	
Meatball BI	MB-0001
Meatball	MB-0004
C	
Sausage	MB-0005

The full code:

How to Set Data with Null Values in FlexCharts?

Context

We use a library named Wicked Charts for server side validation of the chart. Unfortunately this library doesn't support null values, even though they are valid for Highcharts.

Workaround

Replacing the null values by empty maps ("[ ]" in Groovy) will pass the validation successfully, and the final Highchart will look as if nulls had been provided.

A typical FlexChart that would fail due to nulls looks like this:

And this is the result after applying the workaround:

How to Get Quote's Effective and Expiry Dates using Calculation Logic

To fetch the 'Effective Date' & 'Expiry Date' from the Quote's Header Info using a calculation logic, use:

```
quoteProcessor.getQuoteView().targetDate
quoteProcessor.getQuoteView().expiryDate
```

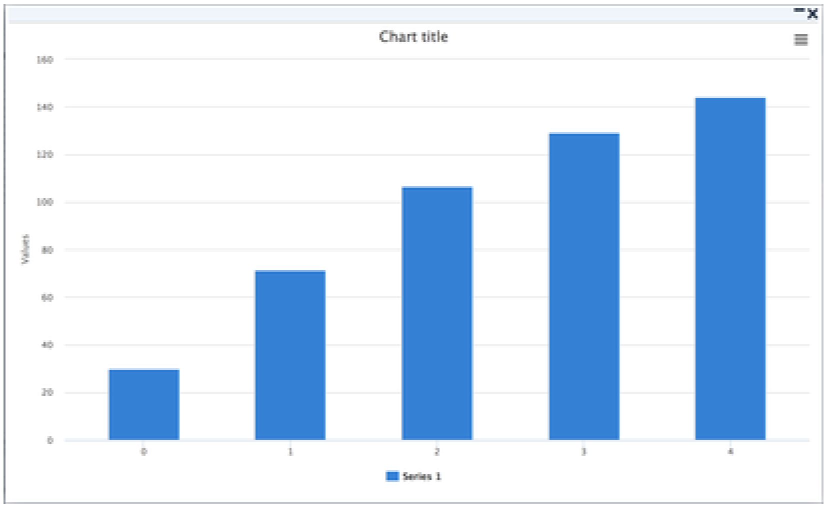
Also, there is a function to get the TargetDate on the line item (but not for the expiry). You can read it in the pre-phase of the header logic and store it into hidden input of the line items.

```
def effectiveDate = quoteProcessor.getQuoteView().targetDate
for (lineItemMap in quoteProcessor.getQuoteView().lineItems) {
if (lineItemMap.folder) continue//skip folders
quoteProcessor.addOrUpdateInput(lineItemMap.lineId,
["name": "effectiveDate",
 "label": "Effective Date",
 "type" : InputType.HIDDEN,
 "required" : false,
 "readOnly" : true,
"value": effectiveDate
]
 )
}
```

FlexChart Examples

- Column Chart, using Default Template
- Chart with full definition in a Template
- Line chart with Definition which overrides the Template
- Waterfall chart
- Column Series
- Column and Spline chart
- Overlapping Area and Column chart

COLUMN CHART, USING DEFAULT TEMPLATE

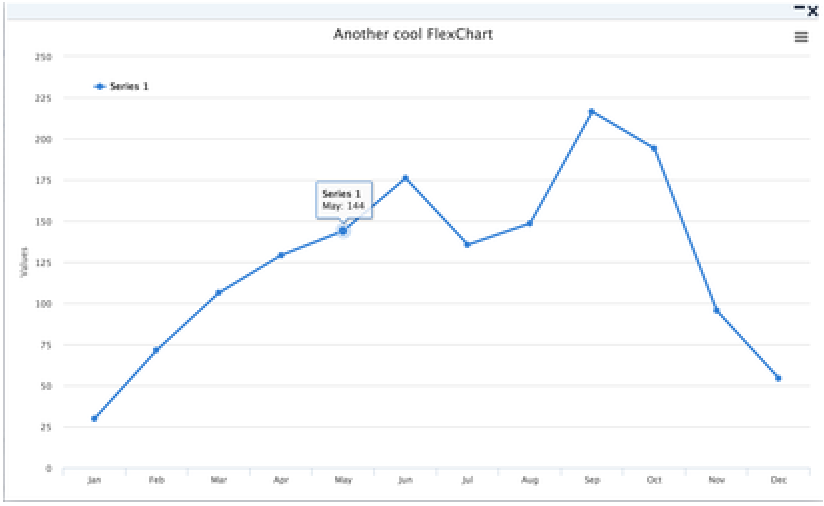


```
def data = [
 series: [
     [
   data: [29.9, 71.5, 106.4, 129.2, 144.0]
  ]
    ]
]

api.buildFlexChart(data)
```

Template "Default"

```
// The default template specifies options which are applied if no
explicit template was selected
// Still gets merged with the base template

{
     "chart": {
         "type": "column"
     },
     "tooltip": {
         "formatter": function() {
   return 'The value for <b>'+ this.x +
    '</b> is <b>'+ this.y +'</b>';
   }
     }
}
```

CHART WITH FULL DEFINITION IN A TEMPLATE

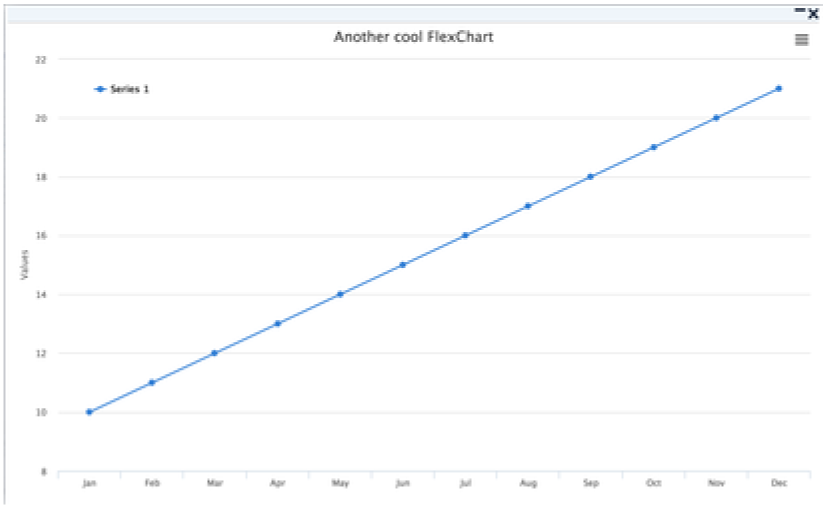


```
api.buildFlexChart("FlexChart_tests_T001", "{}")
```

Template FlexChart_tests_T001

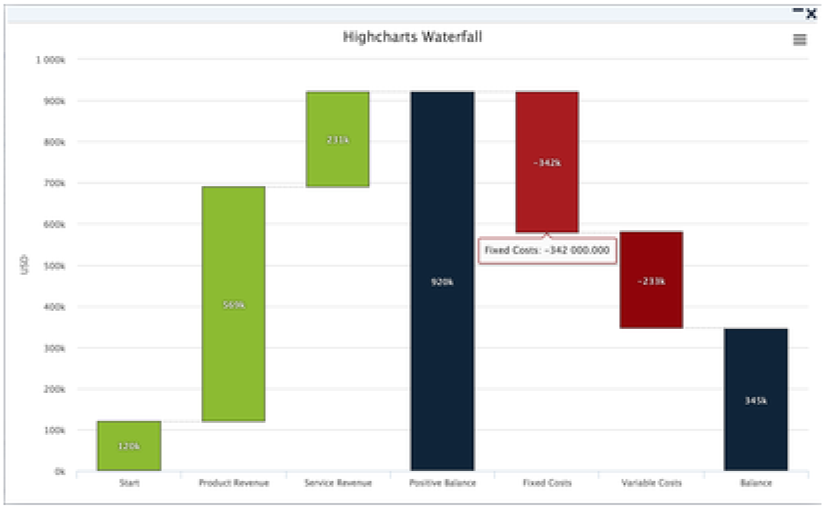
```
{
        "chart": {
            "type": "line"
        },
        "title": {
            "text": "Another cool FlexChart"
        },
        "xAxis": {
            "categories": ["Jan", "Feb", "Mar", "Apr", "May", "Jun",
"Jul", "Aug", "Sep", "Oct", "Nov", "Dec"]
        },
        "yAxis": {
        },
        "legend": {
            "layout": "vertical",
            "floating": true,
            "backgroundColor": "#FFFFFF",
            "align": "left",
            "verticalAlign": "top",
            "y": 60,
            "x": 90
        },
        "tooltip": {
            "formatter": function() {
                return '<b>'+ this.series.name +'</b><br/>'+
                    this.x +': '+ this.y;
            }
        },
        "plotOptions": {
        },
        "series": [{
            "data": [29.9, 71.5, 106.4, 129.2, 144.0, 176.0, 135.6,
148.5, 216.4, 194.1, 95.6, 54.4]
        }]
    }
```

LINE CHART WITH DEFINITION WHICH OVERRIDES THE TEMPLATE



```
def data = [
 series: [
     [
         data: [10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21]
        ]
    ]
]

api.buildFlexChart("FlexChart_tests_T001", data)
```

WATERFALL CHART

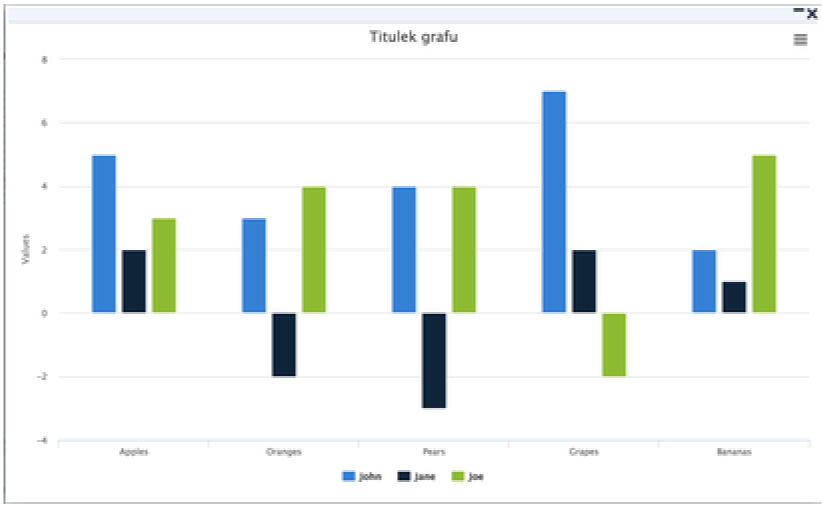


```
def data = [
        title: [
            text: 'Highcharts Waterfall'
        ],

        yAxis: [
            title: [
                text: 'USD'
            ]
        ],

        series: [[
            data: [[
                name: 'Start',
                y: 120000
            ], [
                name: 'Product Revenue',
                y: 569000
            ], [
                name: 'Service Revenue',
                y: 231000
            ], [
                name: 'Positive Balance',
                isIntermediateSum: true,
                color: "#0d233a"
            ], [
                name: 'Fixed Costs',
                y: -342000
            ], [
                name: 'Variable Costs',
                y: -233000
            ], [
                name: 'Balance',
                isSum: true,
                color: "#0d233a"
            ]]
        ]]
    ]

api.buildFlexChart("hc_wf_base", data)
```

Template "hc_wf_base"

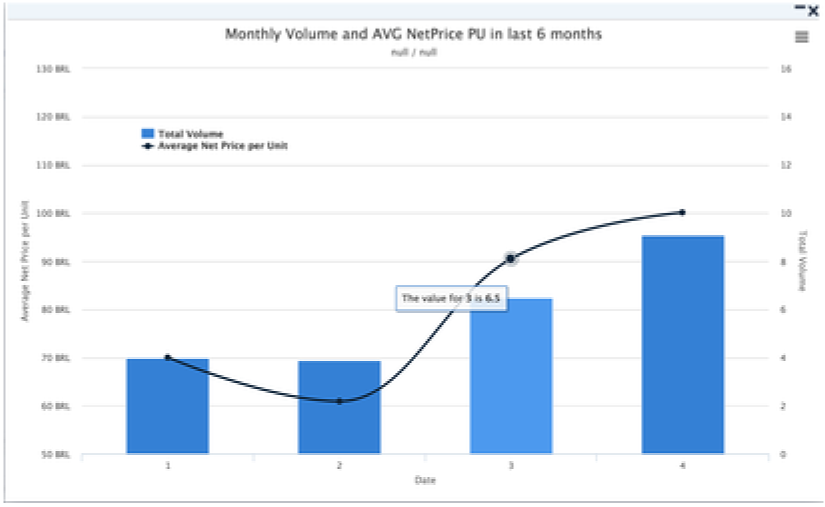
```
{
        "chart": {
            "type": "waterfall"
        },

        "xAxis": {
            "type": "category"
        },

        "yAxis": {
        "title": {
            "text": "K USD"
        }
    },
        "legend": {
            "enabled": false
        },

        "tooltip": {
          "headerFormat": "",
          "pointFormat": "{point.name}: {point.y:,.3f}"
        },

        "plotOptions": {
          "waterfall": {
              "upColor": "#8bbc21",
              "color": "#910000",
              "dataLabels": {
                  "enabled": true,
                  "formatter": function () {
                      return Highcharts.numberFormat(this.y / 1000, 0,
',') + 'k';
                  },
                  "style": {
                      "color": "#FFFFFF",
                      "fontWeight": "bold",
                      "textShadow": "0px 0px 3px black"
                  }
              },
              "pointPadding": 0
         }
        }
}
```

COLUMN SERIES



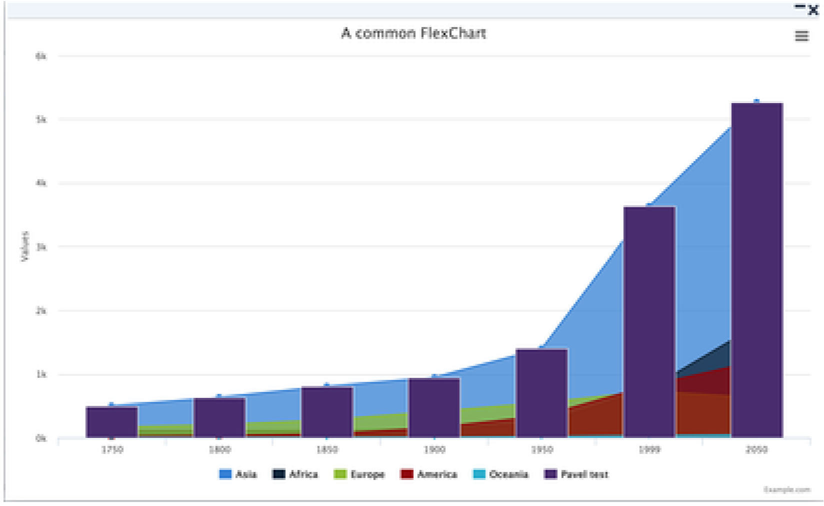
```
def fChart = [
        chart: [
            type: 'column'
        ],

        title: [
            text: 'Chart Title'
        ],

  xAxis: [
            categories: ['Apples', 'Oranges', 'Pears', 'Grapes',
'Bananas']
        ],

  series: [[
            name: 'John',
            data: [5, 3, 4, 7, 2]
        ], [
            name: 'Jane',
            data: [2, -2, -3, 2, 1]
        ], [
            name: 'Joe',
            data: [3, 4, 4, -2, 5]
        ]]
    ]

api.buildFlexChart(fChart)
```

COLUMN AND SPLINE CHART



```
if(api.syntaxCheck) return

def customerId = api.getElement("CustomerId")
def customerName = api.getElement("CustomerName")
def specCode = api.getElement("SpecCode")
def specCodeDescription = api.getElement("SpecCodeDescription")

api.buildFlexChart([
  chart: [
    type: "column"
  ],

  title: [
    text: "Monthly Volume and AVG NetPrice PU in the last 6 months"
  ],
  subtitle: [
    text: customerName + "  /  " + specCodeDescription
  ],
        xAxis: [
            title: [
                text: 'Date'
            ],
          categories: ["1", "2", "3", "4"],
          //categories: getNetPriceHistory(customerId, specCode)[0]
        ],
        yAxis: [[
          title: [
                text: 'Average Net Price per Unit'
          ],
          labels: [
            format: '{value} BRL'
          ],
          ],
          [
            title: [
                  text: 'Total Volume'
            ],
          opposite: true
          ],
        ],

     tooltip: [
         shared: true
     ],

     legend: [
            layout: 'vertical',
            align: 'left',
            x: 150,
            verticalAlign: 'top',
            y: 120,
            floating: true
            //backgroundColor: (Highcharts.theme &&
Highcharts.theme.legendBackgroundColor) || '#FFFFFF'
     ],

        series: [
          [
            name: "Total Volume",
           type: 'column',
            yAxis: 1,
            data: [4.0, 3.9, 6.5, 9.1]
            //data: getNetPriceHistory(customerId, specCode)[1]
          ],
          [
            name: "Average Net Price per Unit",
           type: 'spline',
            data: [70.0, 60.9, 90.5, 100.1],
//            data: getNetPriceHistory(customerId, specCode)[2],
            tooltip: [
             valueSuffix: ' BRL'
         ]
          ]
          ]
])
```

OVERLAPPING AREA AND COLUMN CHART



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


def definition = [

   chart: [
     type: "area",
     //step: 'left'
   ],

  xAxis: [
        categories: ['1750', '1800', '1850', '1900', '1950', '1999',
'2050'],

    ],


   title: [
     text: "A common FlexChart"
   ],

  series: [
    [
        name: 'Asia',
        type: "area",

        data: [502, 635, 809, 947, 1402, 3634, 5268]
    ], [
        name: 'Africa',
        type: "area",
        data: [106, 107, 111, 133, 221, 767, 1766]
    ], [
        name: 'Europe',
        type: "area",
        data: [163, 203, 276, 408, 547, 729, 628]
    ], [
        name: 'America',
        type: "area",
        data: [18, 31, 54, 156, 339, 818, 1201]
    ], [
        name: 'Oceania',
        type: "area",
        data: [2, 2, 2, 6, 13, 30, 46]
    ],

    [
        name: 'Pavel test',
        type: "column",
        data: [502, 635, 809, 947, 1402, 3634, 5268]
    ]

  ]
```

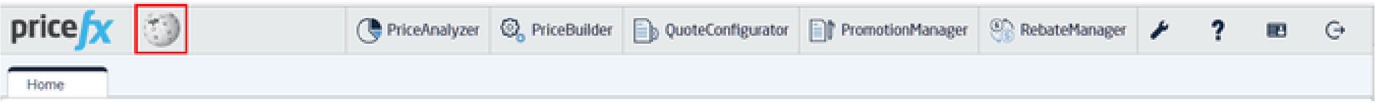
```
]

api.buildFlexChart("AreaTemplate", definition)
```

How to Add Your Own Button to Navigation

You can add a custom button to the main navigation bar, next to the logo. This button can lead to any URL of your choice.

To add the button:

1. Go to Configuration > Advanced Configuration Options.
2. Create a new option called 'extraNavigationHtml'. As a value enter an HTML code containing the <a href> tag.
3. Save the option and refresh the application.



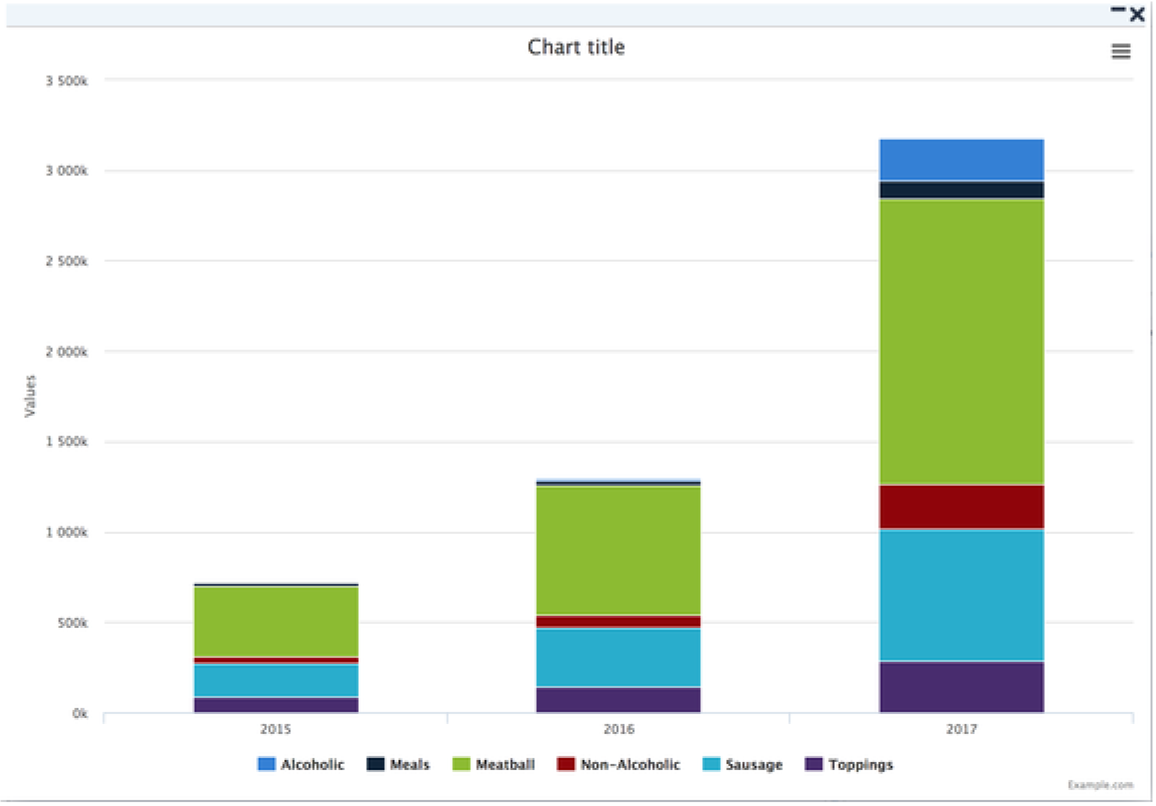
```
<a target="_blank" href="https://en.wikipedia.org"><div
style="width:30px;height:30px; background-size: 30px 30px;
background-image:url(https://upload.wikimedia.org/wikipedia/en/8/80/Wiki
pedia-logo-v2.svg)"></div></a>
```

Stacked Column Chart (using the Flex Chart)

You can use this code on top of any Datamart, just adjust the last line of the code with the name of your:

- Datamart Name
- Dimension column name, used for X axis values.
- Dimension column name, used for the series, which forms the stack.
- Measure column name and operation, if you want to summarize the measure, or calculate the average, etc.

The result then looks like this:



Full code example, which creates the Flex Chart

```
def stackedColumnChartDataFromDatamart(datamartTableName,
dimensionColumnNameForX, dimensionColumnNameForStack, measureExpression)
{
    def dmCtx = api.getDatamartContext()
    def salesDM = dmCtx.getDatamart(datamartTableName)
    def datamartQuery = dmCtx.newQuery(salesDM, true)
    datamartQuery.select(dimensionColumnNameForX, "dimX")
    datamartQuery.select(dimensionColumnNameForStack, "dimStacked")
    datamartQuery.select(measureExpression, "measure")
    datamartQuery.orderBy(dimensionColumnNameForX,
dimensionColumnNameForStack)
    def result = dmCtx.executeQuery(datamartQuery)

    def rows = []
    for (def it = result.getData().iterator(); it.hasNext();) {
        def row = it.next()
        rows << [dimX: (row.dimX), dimStacked: (row.dimStacked),
measure: (row.measure)]
    }

    return rows
}

def createStackedColumnChart(rows) {
    def serieNames = rows.collect { row -> row.dimStacked }.unique()
    def xValues = rows.collect { row -> row.dimX }.unique()

    //populate data array
    def values = [:]
    for (serieName in serieNames) for (xValue in xValues)
values[serieName, xValue] = 0
    for (row in rows) values[row.dimStacked, row.dimX] = row.measure

    //create chart data series
    def series = []
    for (serieName in serieNames) {
        def data = xValues.collect { xValue -> [name: (xValue), y:
(values[serieName, xValue])] }
        def serie = [name: (serieName), data: (data)]
        series << serie
    }

    //chart definition
    def chart = [
            chart      : [
                    type: 'column'
            ],
            xAxis      : [
                    categories: (xValues)
            ],
            plotOptions: [
                    column: [
                            stacking: 'normal'
                    ]
            ],
            series     : (series)
    ]


    return api.buildFlexChart(chart)
}

if (api.isSyntaxCheck()) return

return createStackedColumnChart(
        stackedColumnChartDataFromDatamart("Sales_Data",
"Invoice_DateYear", "ProductGroup", "SUM(Quantity)")
```

```
)
```

How to Set Value Labels in Drop-Down in Header Logic

There's a method `Object option(String entryName, List<Object> options, Map<String,Object> labels)` which allows you to set values' labels in the drop-down. To set these labels when adding the drop-down in a quote header logic, use the following examples.

This is a header-level drop-down with simple options (just the list of values):

```
def endCustomerList = [
 "customer1",
 "customer2",
 "customer3"
]

quoteProcessor.addOrUpdateInput("ROOT", [
 "name"    : "endCustomer",
 "label"   : "End Customer",
 "type"    : InputType.OPTION,
 "valueOptions"  : endCustomerList,
])
```

If you want to add labels, you need to assign a map with key "labels" to a field "parameterConfig". The "labels" will contain your map with key-value pairs, where key is the name of the option and value is the label.

```
def mapWithLabels = [
 "customer1" : "First Customer",
 "customer2" : "Second Customer",
 "customer3" : "Third Customer"
]

quoteProcessor.addOrUpdateInput("ROOT", [
 "name"     : "endCustomer",
 "label"    : "End Customer",
 "type"     : InputType.OPTION,
 "valueOptions"   : endCustomerList,
 "parameterConfig" : [
  "labels": mapWithLabels
 ]
])
```

How to Access QC Line Outputs in Workflow Template

To access QuoteConfigurator item level outputs in the template 'quote_creationWorkflowStepAssigned' for a document creation workflow:

- Access the quote object in the template.
- Access quote.lineItems.

- Iterate through it and access 'outputs'.
- Iterate through it and display the results.

```
<html>
<body>
The following items on the quote $quote.uniqueName$ - $quote.label$ were
marked as won:

$quote.lineItems:{ item |
 $if(item.priceRecordId)$
        $item.sku$
 $endif$
}$
<p> Created by: $quote.createdByName$ </p>
</body>
</html>
```

How to Display Large Number of Data Points in FlexChart Scatter Plot

The FlexCharts are configured using the underlying Highcharts framework's options. For large data sets, Highcharts features the so-called turbo mode, to speed up the display. By default, the number of points above which this mode kicks in is set to 1000 data points, but this value can be modified in the options. One typical way of providing a data point in Highcharts is a Map, e.g. in Groovy ['x': 1.5, 'y':0.2]. However, if the turbo mode kicks in, Maps are no longer supported and the chart might remain blank. The solution is to provide the points as a simple array, e.g. replace the previous expression by [1.5, 0.2]. It is also possible to raise the turboThreshold value, or to simply disable the turbo, but this may have a noticeable performance impact.

The below code shows a very simple FlexChart scatter plot that uses Maps for data points and will not display because the data array size is 1001. The commented out code shows the two ways of fixing it.

Note: One reason why option 1 may not work for you is the case when the data point's tooltip must display additional values, such as text values or additional measures.

How to Add a Map to Dashboard

You can add a map to your dashboard, so that you can display regional data in a graphical form.

All you have to do is to create a Highmaps map widget and use it in your dashboard. The widget is defined by a calculation logic using the `api.buildHighmap` method. The data can come either from PriceAnalyzer or other storages or can be created on the fly.
Refer to the Highmaps API reference to explore the full variety of settings. At the very least you need to provide a `chart.map` value, `series[n].joinBy` value and `series[n].data`.

The `joinBy` array tells Highmaps how to match the map definition with the data. Typically for international maps, the country name is the '`name`' field, which matches the '`country`' field of the data in our the example above. There are also '`iso-a2`' and '`iso-a3`' fields, which feature the 2- and 3-character ISO standards, and many more. Exploring the JavaScript definitions on the Highmaps - Map Collection page should help understanding how this works.

The following example is based on Color axis and data labels with a 1 to 1 translation from JavaScript (as found in "View Options" on the linked page) to Groovy:

> If you cannot find an example of an existing logic that uses a chart type that you require, it should be fairly easy to adapt one of the Highmaps Demos. Select your chart and click on the "View Options" button and you will see the JS source code, including a `Highcharts.mapChart('container', {...})` call. This `{...}` content is the `definition` map. You will have to convert it from JS to Groovy. This always implies replacing the curly brackets by square brackets (`{} -> []`). But you may also sometimes encounter code like `Highcharts.color(colors[5]).brighten(0.2).get()` which you should replace by your own hard coded colors, or `formatter: function () {...}` which you would have to replace by simpler pattern formatter `format: '{point.name}: {point.y:.1f}%'`.

⚠ Currently, not all maps of the extensive Highmaps collection are supported (bundled) – see the list below. If you are missing a map, let us know (open a support ticket) and we will do our best to add it in the next patch release.

World and Continents:

- /custom/africa
- /custom/asia
- /custom/europe
- /custom/european-union
- /custom/oceania
- /custom/central-america
- /custom/north-america
- /custom/north-america-no-central
- /custom/usa-and-canada
- /custom/world
- /custom/world-continents
- /custom/world-eckert3
- /custom/world-palestine
- /custom/world-robinson

Countries:

Country	Map
Australia	countries/au/au-all[1]
Canada	countries/ca/ca-all[1]
China	countries/cn/cn-all[1] [2] countries/cn/custom/cn-all-sar[1] [2] countries/cn/custom/cn-all-sar-taiwan[1] [2]
Czech	countries/cz/cz-all[1]
Denmark	countries/dk/dk-all[1]
Finland	countries/fi/fi-all[1]
France	countries/fr/fr-all[1]
Germany	countries/de/de-all[1]
Hungary	countries/hu/hu-all[1] [2]
Italy	countries/it/it-all[1]
Japan	countries/jp/jp-all[1] [2]
Netherlands	countries/nl/nl-all[1]
New Zealand	countries/nz/nz-all[1]
Poland	countries/pl/pl-all[1] [2]
South Korea	countries/kr/kr-all[1]
Spain	countries/es/es-all[1]
Sweden	countries/se/se-all[1]
Switzerland	countries/ch/ch-all[1]
UK	countries/gb/gb-all[1] countries/gb/custom/gb-countries[1]
US	countries/us/us-all countries/us/custom/us-all-territories[1]

[1] since El Presidente PR4 (3.6.4)

[2] minor character encoding issues remain

How to Delete Deployed Calculation Flow

Like in the UI, deployed calculation flows (CF) cannot be deleted by users. You can only create a new empty CF with the same name and the CF processor will delete both.

```
import net.pricefx.automation.MyClient
import net.pricefx.automation.TypeCode
import net.pricefx.automation.automations.CalculationFlowsAutomation
import net.pricefx.automation.automations.GenericAutomation
import net.pricefx.integration.api.client.builder.FilterCriteriaBuilder
import net.pricefx.integration.api.client.model.CalculationFlow
import net.pricefx.integration.api.client.model.FilterCriteria

import
staticnet.pricefx.integration.api.client.builder.CriteriaBuilder.equal;
// TODO - you need to rewrite this code to use proper PriceFxClient - to
link to your specific environment
def clients = MyClient.getClients("QA", [
"companyname",

], false);
// TODO - change CF name here
def CF_Name_TO_BE_DELETED = "test"

FilterCriteriacriteriaDeployed = newFilterCriteriaBuilder()
.and()
.with(equal("uniqueName", CF_Name_TO_BE_DELETED))
.with(equal("draft", false))
.create();
FilterCriteria criteriaDraft = newFilterCriteriaBuilder()
.and()
.with(equal("uniqueName", CF_Name_TO_BE_DELETED))
.with(equal("draft", true))
.create();

for (MyClient client : clients) {

println(CF_Name_TO_BE_DELETED + "tobedeleted")

def cfDraft = CalculationFlowsAutomation.fetchCalculationFlow(client,
criteriaDraft)
def cfDeployed = CalculationFlowsAutomation.fetchCalculationFlow(client,
criteriaDeployed)
if (cfDraft) {
CalculationFlowsAutomation.deleteCalculationFlow(client, cfDraft)
}
if (cfDeployed) {
cfDeployed.setFlowId(null)
 cfDeployed.setConfiguration(null)
CalculationFlowsAutomation.addCalculationFlow(client, cfDeployed)
}
}
```

How to Pass Header Input Values to Inline Configurator

If you need to pass Quote, Contract or Rebate Agreement header field values to the inline configurator logic, you can use the `api.contextTypedId()` method. This way, you can obtain the TypedId and further work with it (e.g., fetch a Quote).

How to Mass Update Quotes with a Configurator

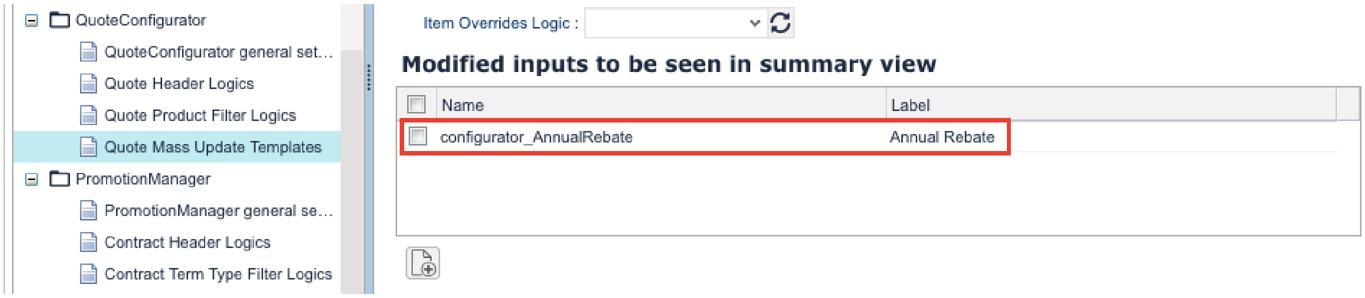
If you have in your implementation a **header-level configurator** with user inputs that you want to modify during a quote mass update, this section helps you configure a solution.

This is a sample logic that defines a configurator:

The quote header logic will look similar to this one:

Now you must create a header input mass update logic. This logic will generate an input parameter in the Set Parameters step of the Mass Update dialog. If you need to update the input with a static value, simply replace the "r" in the below code with the required value:

To be able to see the inner configurator's input on the mass update Summary page, add it to the Modified Inputs table in the following format: configuratorName___inputName (note that there must be **three underscores**).


QuoteConfigurator
QuoteConfigurator general set...
Quote Header Logics
Quote Product Filter Logics
Quote Mass Update Templates
PromotionManager
PromotionManager general se...
Contract Header Logics
Contract Term Type Filter Logics
Item Overrides Logic :

Modified inputs to be seen in summary view

□	Name	Label
□	configurator_AnnualRebate	Annual Rebate

> See also: Configurators and Quote Mass Update.

How to Make Home Panels Always Visible

Two sections of the Home page, 'My ToDo List' and 'My Pending Approvals', are displayed only if there is a content to display.

If you want to have them always shown (even if empty), set the following keys in Advanced Configuration Options to `true`:

- `alwaysShowMyPendingApprovals`
- `alwaysShowMyToDos`

ℹ The settings made here using the Advanced Configuration Options have **lower** priority than settings made in Configuration > Home Page (where you control visibility of the Home page panels).

How to Deal with PA Data Table's UI Responsiveness?

This section summarizes some of the performance issues of PriceAnalyzer Data Tables and their possible resolutions.

Note that this article refers to the number of *displayed* rows in the Data Table, typically from a few hundreds to a few thousands.

General Notes

- For Data Tables, there is a hardcoded limit of maximum 10 000 displayed rows (and maximum of 8 grouping levels). The number of rows in the underlying Datamart is not so relevant. But of course if there are millions of rows, at some point the backend query may also have an impact.
- Unless the user has a high end hardware (quad core CPU and 8+ GB of RAM), the UI will become very slow above 4 group by levels with 2000+ rows.

Issues

DATA TABLES ARE SLOW TO LOAD

Issue: Data Tables may become really slow to load, and this is due to SmartGWT's processing of the grouping levels (well noticeable from 5 to 8 levels). (SmartGWT is a 3rd party underlying UI framework.) It is quite common to experience loading 4000+ rows for about one minute on a fast machine (i7 quad with 16GB of RAM) in any browser.

Resolution: See Conclusions below.

UNRESPONSIVE APPLICATION IN FIREFOX

Issue: Once a Data Table has been displayed, the application may become very unresponsive in Firefox – for about as long as it took for the table to load.

Resolution: It is recommend to open the task manager, check out Firefox's CPU usage and leave the browser alone until it goes down. The experience shows that when trying to interact with the browser, it will take much longer to recover. Waiting is well worth it as Firefox then performs quite well once it has fully recovered.

UI GENERALLY LESS RESPONSIVE, EVEN AFTER SWITCHING TABS

Issue: Once a Data Table has been displayed, the UI may become less responsive – even after switching to a different PFX tab. This behavior was observed on Mac with Chrome and Safari; presumably Chrome on Windows will perform similarly. It happens when 1000+ rows are displayed, either when the table is totally ungrouped, or when it is grouped with most or all of its nodes expanded.

Resolution: See Conclusions below.

Conclusions

To deal with the issues above, try the following:

- Make sure to filter the data if more group by levels are required.
- Check out the rollups functionality and use it where appropriate. Unless the tree view (the Group button below the grid) is required, rollups will solve these limitations.
- Save the Data Table in a grouped state. This is strongly recommended if the grouping functionality is indeed required and the performance impact becomes noticeable. The table will take longer to load but will have less of an impact if the tab remains open.

How to Get Full Log File for Specific Node

It may happen that a scheduled job fails with errors on a specific node on your partition. To be able to troubleshoot the issue, you need to get a full log file for that node. If you select the option 'View Logs' or 'View Logs (Full)' in Administration, you will get the logs from the node where you are logged in.

However, some calculations (like Rebate Calculation or PADataLoad) can run on a different (background processing) nodes, which means you do not get the log info for the calculation you are trying to debug.

To get a log file for a different node:

1. Find out which node you need the logs for, either from the error message if you got any (such as: Rebate Calculation scheduled job failed with errors on "node2" for xyz partition) or from the Job/Task Tracking section in Administration where you find your process.
2. Select 'View Logs' in Administration.
3. Manually change the "nodeX" in the URL to the one you want and resubmit.

For example, URL formats for a staging environment look like this:

- Log tail: https://staging.pricefx.eu/logtail/logtail.html?partition=PARTITIONNAME&node=node1.staging-ca
- Full log (TXT download): https://staging.pricefx.eu/pricefx/PARTITIONNAME/redirect/node1.staging-ca/viewlog/full

How to Make the Simple Filter Search for Exact Values

The default behavior of simple filters is that they search for all entries, which **contain** the string that you typed. However, if you have a lot of data, this will significantly slow down your searches. If you know that in most cases you will be happy with a search result that **exactly** matches your search string, you can change the application's configuration (per partition).

To change the default configuration:

1. Go to Administration > Configuration > Advanced Configuration Options.
2. Add the property "defaultTextMatchStyle" and set its value to "exact". The property accepts values "exact" and "startsWith".
3. Reload the application.

Note: This change will affect all tables (except for those that already had special behaviour) in the application. It is not possible to apply this behavior only to selected tables.

How to Hide Download DOCX Icon

It is possible to download objects (e.g. Quotes, Contracts, Rebate Records, Price Records) in the DOCX format (using a predefined template). This 'Download Word' icon is shown under the data tables by default. If you wish to hide it, go to Administration > Configuration > Advanced Configuration Options and set the "disabledDocxDownload" property value to "true".

How to Add Image to Product Detail

If you want to have a product image as part of the product details, you can add an image as a content of a portlet in the Product Detail View.

1. Go to Master Data > Products.
2. Select a product and click View/edit details.
3. Add a new portlet of the Product Image category.
4. Click the Upload button and choose an image file. You can replace the image by uploading a new one or delete it.

Once the image is uploaded, it can be retrieved via a URL for authenticated users in the same partition. The URL will have the following pattern:

`https://pricefx.eu/pricefx/<partition>/productimages.get/<sku>`

You can further specify the image size in the URL so that the image is not retrieved in its full size and resized in the browser. This is useful if you want to include images of specified sizes in a Configurator.

Type	Description	Example
Thumbnail	Resizes the image to the size specified in the PFX server settings.	`/<sku>?predefinedSize=thumbnail`
Fixed Width	Resizes the height accordingly to the aspect ratio of the original image.	`/<sku>?width=100&height=200&type=FIXED_WIDTH`
Fixed Height	Resizes the width accordingly to the aspect ratio of the original image.	`/<sku>?width=100&height=200&type=FIXED_HEIGHT`
Boxed Size Crop	Resizes the image to fill the specified box size – the image is cropped.	`/<sku>?width=174&height=198&type=BOXED_SIZE_CROP`
Boxed Size Fit	Resizes the image keeping the aspect ratio and fitting the longest size to the specified box.	`/<sku>?width=174&height=198&type=BOXED_SIZE_FIT`

The resized images are cached using the standard HTTP caching making them quicker to load next time.

Uploading Image Using API

Integration engineers can use the API to upload images of products, so that the integration route can upload product images automatically.

The API consumer will have to:

- Create an upload slot first by calling **uploadmanager.newuploadslot**.
- Then upload the product image by calling **productimages.upload/slot ID/sku**.
- Finally delete the uploading slot by calling **uploadmanager.deleteslot/slot ID**.

How to Set Language and Locale via URL

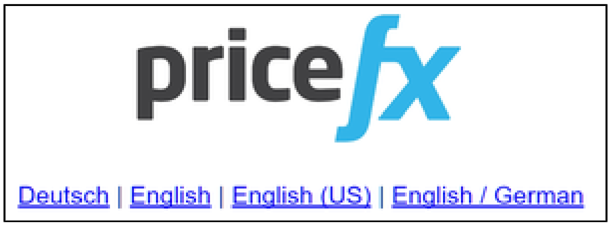
On a landing page for a PFX instance, you can use these parameters in links to different language versions:

- `locale` – Sets the format of dates and numbers.
- `uiLanguage` – Sets the language of the application. If not specified, uses the language specified by `locale`

As a value for both parameters, use the ISO language code.

For example, the last link on the screenshot below looks like this: `https://test.pricefx.eu/priceFxWeb.html?locale=de&uiLanguage=en#loginPage`

It keeps the UI in English, while dates and numbers are formatted according to German.



How to identify whether a quote is duplicated, or revised, or new in Quote Header Logic?

def version = quoteProcessor.getQuoteView().version

def prevRev = quoteProcessor.getQuoteView().prevRev

New Quote

version != null or (version == 3 && prevRev == null)

Revision

prevRev != null

Duplicated Quote

version != null and version != 1 and version != 3

Inputs Behavior in Quote Revisions

Upon a quote revision not only all items are re-calculated but also inputs are re-generated (by running them once in the syntax check mode; the same way as it is done when adding a product).

When inputs with the same name are generated, the values are carried over from the old version. Inputs that do not exist anymore are discarded and new inputs are added. The reason why this is done is that the formula or other things may have changed between the versions. Hence you need to do a full "at the current time" sanity check, otherwise you would carry on with an old format that may be outdated already.

When you have inputs in the header formula, it becomes tricky. As described above, the input generation is on the item level and drops all non-generated inputs as it considers them outdated.

✓ The conclusion is that ideally the line item logic should create these inputs. Then they will stay.

Example

Issue	Resolution

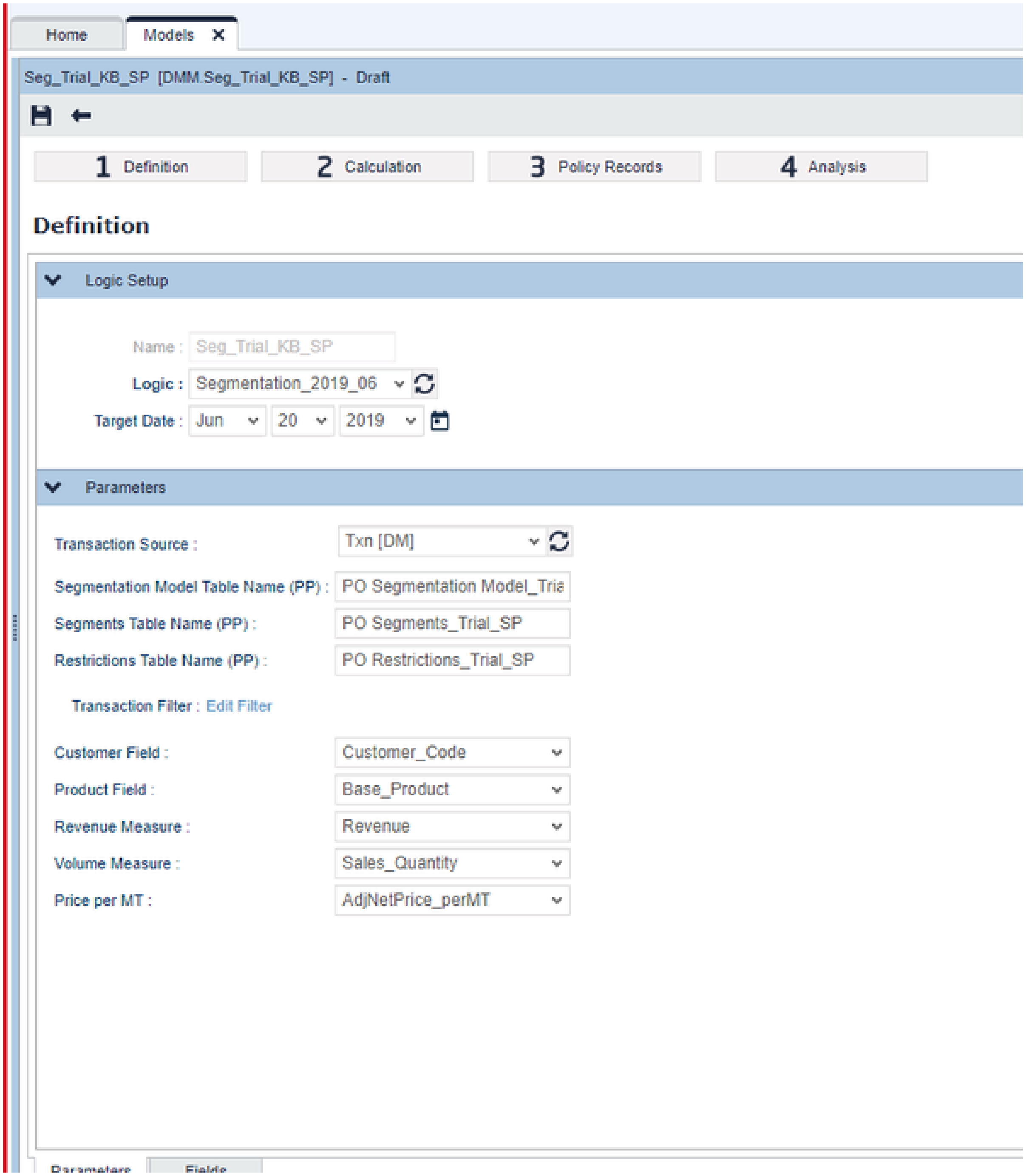
If you create an input for a line item in the header logic (through `quoteProcessor.addOrUpdateInput()`) and create a quote revision, the values from the inputs disappear.
This happens only for line items; when added to ROOT, the inputs preserve their values.
Header Logic Snippet
`for (lineItemMap in`
`quoteProcessor.getQuoteView().lineItems) {`
`    if (lineItemMap.folder) continue  //skip folders`
`    quoteProcessor.addOrUpdateInput(lineItemMap.lineId,`
`            ["name"     : "SampleInput",`
`             "type"     : InputType.STRINGUSERENTRY,`
`             "required" : true`
`            ]`
`    )`
`}`
	The desired behavior someh
	The solution is to have an in
	The code snippet here can b to the calculation logic for a
	`api.string`
	Then the header logic can e
	With this change, the input e

How to Configure Segmentation and Optimization Models

To configure a model, follow these steps:

1. Create Segmentation:
    a. Set Names
    b. Assign Logic (opt. 06)
    c. Set Transaction Sources
    d. Set unique names of the 3 configuration tables, starting with "PO ", so that they correspond to current model

   e. Set Filter
   f. Save
   g. Assign Fields (Customer , Product …..)
   h. Save
2. Calculation:
   a. Price Drivers => Minimum Threshold + Truncate and Calculate. To define what are Price Drivers – importance of Attributes. "Job/Task Tracking". Back to drivers. It is important to refresh.
   b. Segmentation Level => Unselect all and Select the appropriate dimensions in the right sequence (order is important, unchecked items in between have no influence) => Set the transactions Threshold, Truncate and Calculate.
   c. Check “R2” in Data (Refresh before).
   d. Observe segments in Calculation->Data, or in Analysis.
3. Manage «Price Parameters»:
   a. Select one of the 3 tables. Click Activate.
   b. Do the same for the other 2.
   c. One of the tables contains a list of attributes used for segmentation, another a definition of segments, the third one is currently not used.
   d. If older versions of activated tables with same names exist and prevent activation - delete or inactivate the old ones.
4. Copy Seg into opt and modify logic:
   a. Modify Name.
   b. Refresh.
   c. Change Label.
   d. Click on Modify.
   e. Check if Optimization Target date is higher than the validity date for “PO Segmentation Model” and the other two tables.
   f. Select Logic (opt. 06).
   g. Save.
   h. Control the names of the segmentation configuration, starting with "PO " are same as the one activated in the previous step.
   i. Calculation – Segment Scoring. Set number of transactions + Truncate and Calculate.
   j. Calculation – Target Price. Set number of transaction + Calculate (Without Truncate).
   k. Observe segments and floor/target/stretch prices in Calculation->Data, Policy Records, or Analysis.



Seg_Trial_KB_SP [DMM.Seg_Trial_KB_SP] - Draft

Definition

Name : Seg_Trial_KB_SP
Logic : Segmentation_2019_06
Target Date : Jun 20 2019

Transaction Source : Txn [DM]
Segmentation Model Table Name (PP) : PO Segmentation Model_Tria
Segments Table Name (PP) : PO Segments_Trial_SP
Restrictions Table Name (PP) : PO Restrictions_Trial_SP
Transaction Filter : Edit Filter
Customer Field : Customer_Code
Product Field : Base_Product
Revenue Measure : Revenue
Volume Measure : Sales_Quantity
Price per MT : AdjNetPrice_perMT

How to See Users Assigned to a User Group

Audit Logs > Assignment Overview tab will allow you to select User Group and display the related assignments

How to Disable Autocomplete When Entering Values in Advanced Filters

By default, advanced filters use autocomplete, i.e. when you start typing a value in the filter, the list of available values dynamically changes to display only strings with matching initial characters.

This autocomplete behavior can be disabled. To do so, go to Administration > Configuration > Advanced Configuration Options and set the "advancedFilterComboboxDisableAutosuggest" property value to "true".

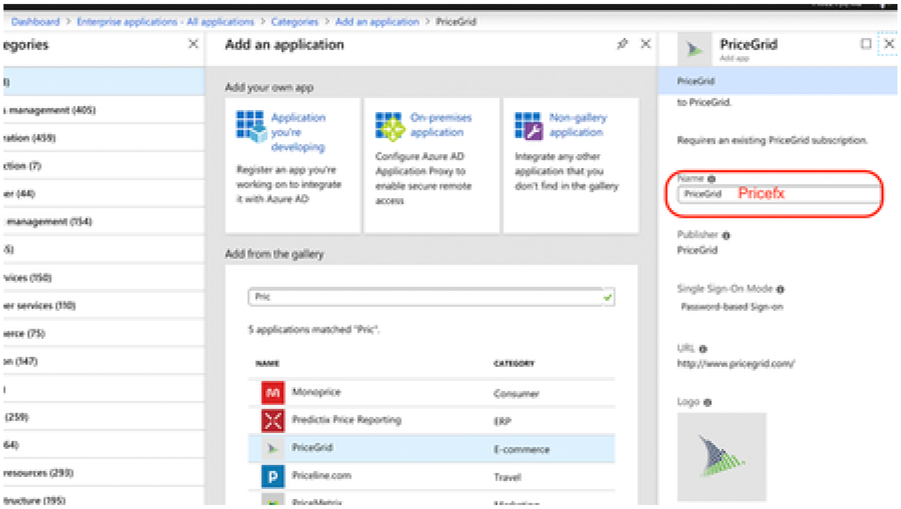
Once it is disabled, you can still type the characters which will restrict the list of available values, but this list gets displayed only after you click the arrow icon in the filter field.

How to Set Up SSO with Azure

The general setup is described in the Pricefx Documentation, please read it first.

How to set up SSO in Price f(x) when Azure Active Directory is used as an identity provider:

1. As Pricefx is not yet available in the app gallery in Azure AD, you need to use this workaround: in the gallery pick an app which is not yet installed and transform/modify it to the Pricefx app. We selected PriceGrid just for the purpose of this example. Change the name to Pricefx:



Dashboard > Enterprise applications - All applications > Categories > Add an application > PriceGrid

egories

management (405)
ration (459)
ction (7)
er (44)
management (154)
5)
vices (150)
er services (110)
erce (75)
on (147)
(259)
64)
resources (293)
tructure (195)

Add an application

Add your own app

Application you're developing
Register an app you're working on to integrate it with Azure AD

On-premises application
Configure Azure AD Application Proxy to enable secure remote access

Non-gallery application
Integrate any other application that you don't find in the gallery

Add from the gallery

Pric

5 applications matched "Pric".

NAME	CATEGORY
Monoprice	Consumer
Predictix Price Reporting	ERP
PriceGrid	E-commerce
Priceline.com	Travel
PriceMetrix	Marketing

PriceGrid
Add app

PriceGrid
to PriceGrid.

Requires an existing PriceGrid subscription.

Name
PriceGrid Pricefx

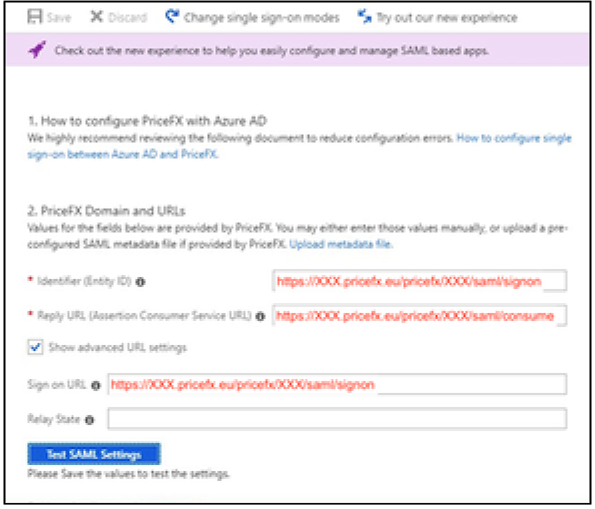
Publisher
PriceGrid

Single Sign-On Mode
Password-based Sign-on

URL
http://www.pricegrid.com/

Logo

2. Proceed and install the app. Once installed, you can change the logo to PFX in the properties. The suggested format is PNG of 215 x 215 px (you can use this file: PFX_Azure_logo.png).
3. Then continue in Azure. Set the following:
   - Identifier (Entity ID): https://xxx.pricefx.eu/pricefx/xxx/saml/signon/
   - Reply URL: https://xxx.pricefx.eu/pricefx/xxx/saml/consume/
   - Sign on URL: https://xxx.pricefx.eu/pricefx/xxx/saml/signon/



Check out the new experience to help you easily configure and manage SAML based apps.

1. How to configure PriceFX with Azure AD
We highly recommend reviewing the following document to reduce configuration errors. How to configure single sign-on between Azure AD and PriceFX.

2. PriceFX Domain and URLs
Values for the fields below are provided by PriceFX. You may either enter those values manually, or upload a pre-configured SAML metadata file if provided by PriceFX. Upload metadata file.

* Identifier (Entity ID): https://XXX.pricefx.eu/pricefx/XXX/saml/signon

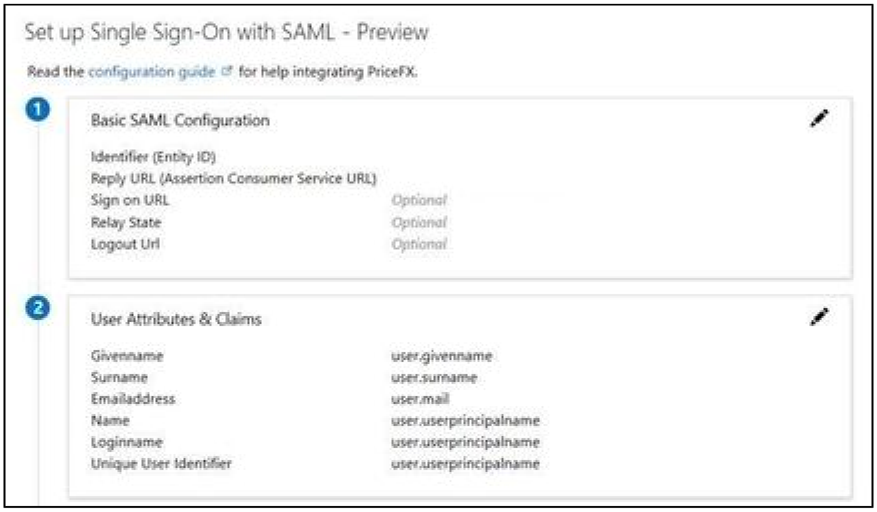
* Reply URL (Assertion Consumer Service URL): https://XXX.pricefx.eu/pricefx/XXX/saml/consume

☑ Show advanced URL settings

Sign on URL: https://XXX.pricefx.eu/pricefx/XXX/saml/signon

Relay State:

Please Save the values to test the settings.



Set up Single Sign-On with SAML - Preview

Read the configuration guide for help integrating PriceFX.

1 Basic SAML Configuration

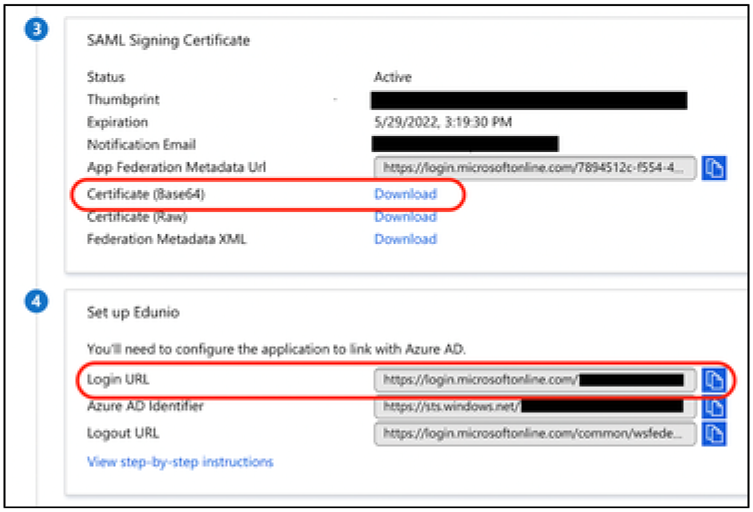
Identifier (Entity ID)	
Reply URL (Assertion Consumer Service URL)	
Sign on URL	Optional
Relay State	Optional
Logout Url	Optional

2 User Attributes & Claims

Givenname	user.givenname
Surname	user.surname
Emailaddress	user.mail
Name	user.userprincipalname
Loginname	user.userprincipalname
Unique User Identifier	user.userprincipalname

4. In Price f(x) > Configuration > SAML Configuration set the following:
   - NameID Mapping: email
   - SAML IdP URL: see in the screenshot below where to get this value in Azure

- IdP Certificate: Base64 encoded



3 SAML Signing Certificate

Status Active
Thumbprint [redacted]
Expiration 5/29/2022, 3:19:30 PM
Notification Email [redacted]
App Federation Metadata Url https://login.microsoftonline.com/7894512c-f554-4...
Certificate (Base64) Download
Certificate (Raw) Download
Federation Metadata XML Download

4 Set up Edunio

You'll need to configure the application to link with Azure AD.

Login URL https://login.microsoftonline.com/[redacted]
Azure AD Identifier https://sts.windows.net/[redacted]
Logout URL https://login.microsoftonline.com/common/wsfede...
View step-by-step instructions

5. Once all this is configured, you need to use the specific URL to log in: https://xxx.pricefx.eu/pricefx/XXX/saml/signon (Only then the SSO login will work. If you go to the regular Pricefx home page URL, e.g. https://xxx.pricefx.eu, there will still be the PFX login screen.) This sign-on link is different for each partition (dev/qa/prod).

Azure AD requires that the identifierUris do **not** have a trailing slash. Therefore the SSO identifier URL in Pricefx is configurable. By default the trailing slash is always added but it can be removed by adding the parameter "trailingIdSlash" : false.

ℹ See also Tutorials for integrating SaaS applications with Azure Active Directory.

Technical Details

- How to Hide Customer Input Parameter for Quote Line Item
- How to Grab a Result for a specific SKU from a LPG
- How to Trigger Partial LPG Calculation from Calculation Flow
- How to Stop the Submit Process
- How to Format Emails via api.sendEmail
- Why NOT to Use 'float' or 'double' Types in Groovy
- How Global Variables are Handled in Second Pass
- How to Set the Number of Displayed Decimals in User Input
- How to Write Formulas
- How to Save Logic that Is Evaluated as Erroneous by Syntax Checker
- How to Check Workflow Step Execution Status
- How to Dynamically Generate Server Request from Dashboard
- How to Create an Interactive Embedded Dashboard
- How to Use Traffic Lights in Data Tables
- Datamart Join Sample Code
- Logic Types Cheatsheet
- How to Get an Email Notification When a Background Job Fails
- How to Group Calculation Results
- How to Set Up Failover Monitoring
- How to Assign a Unique Autogenerated ID to Price Parameters
- How to Show customerId Value from customerGroupEntry Picker in Outputs
- How to Avoid Slow Initial Rendering of Result Matrix?
- How to Set Quote Owner
- How to Authenticate with Swagger
- Function Result Caching Examples

How to Hide Customer Input Parameter for Quote Line Item

If you want to hide the "Customer" input parameter on a quote line item, you cannot use the function `api.customer("attr name")`. Instead, you must use `api.customer("attr name", "customer id")`, with the "customer id" defined. (If it was empty, the system would ask for it.)

Example

```
def customerId = api.input("Customer")
if (customerId) {
 return api.customer("SALESORG",customerId)
} else {
  return
}
```

The `api.input()` does not trigger the input box creation.

How to Grab a Result for a specific SKU from a LPG

If you want to use calculation results of a specific SKU in a Live Price Grid (LPG) in another calculation:

1. Run `api.find` with type `PGI`.
2. A filter on SKU returns all Price Grid Items (lines in a price grid) for a given SKU.

```
def query = api.find("PGI", Filter.equal("sku", mysku))
```

Note:

- *query* can return multiple lines if the SKU is present in multiple LPG’s.
- The structure of `query[i]` depends on the elements of the LPG.
- For the matrix type LPG use the type code "XPGI".

How to Trigger Partial LPG Calculation from Calculation Flow

If you need to calculate only selected items in a LPG, you can trigger the calculation directly from a calculation flow:

```
actionBuilder
.addLivePriceGridAction(label)
.restrictToSkus([sku1,sku2, sku3])
.setCalculate(true)
```

In the `restrictToSkus` method you need to specify the list of skus you want to recalculate. The behavior of the recalculation is the same as in the user action "'calculate" on a single LPG item.

How to Stop the Submit Process

It can be useful to stop the Submit process in case when you find a problem with data (e.g. you discover an error in Quote inputs).

When you hit the Submit button, it executes the workflow logic. But if you raise an exception in the workflow logic, it will stop the submitting and go back to Draft.

```
api.throwException("There is an error in the Quote, please fix it before
submitting.")
```

If this is in a workflow logic for a quote, the message will appear in the Message portlet window of the quote.

How to Format Emails via api.sendEmail

The `sendEmail` function requires three string parameters: the recipient's email address, the subject of the email, and the message of the email.

```
sendEmail(String to, String subject, String message)
```

The email can be formatted via HTML tags and can contain Groovy variables.

Example

```
def user = quote.createdByName
def u = api.find("U", Filter.equal("loginName", user))
def email = u.email[0]

def url =
"https://qa.pricefx.eu/priceFxWeb.html?targetPage=priceShopPage&targetPa
geState=" + quote.typedId + ""
api.sendEmail(email, "Pricefx Quote - " + customerName + " - " +
vendorPart + "", "" + customerName + " ordered " + vendorPart + " at
quantity of " + actualQuantity.toInteger() + ". This is below committed
quantity of " + quotedQuantity + " with tolerance. " + "<br><br>" + "<a
href=" + url + ">View Quote</a>")
```

Why NOT to Use 'float' or 'double' Types in Groovy

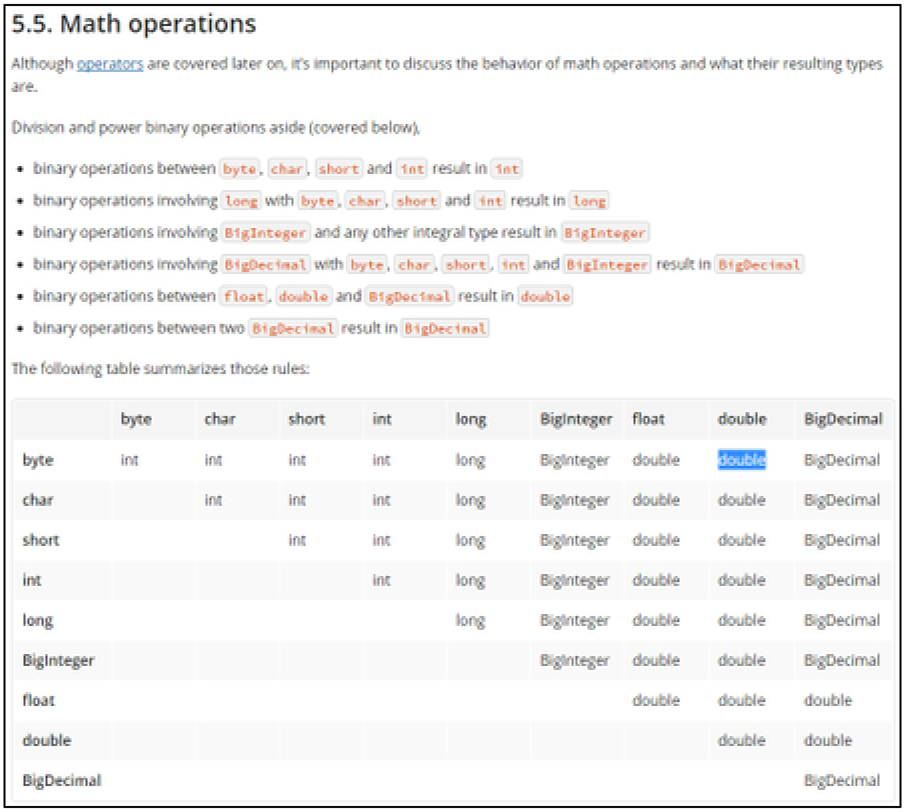
The types 'float' and 'double' should not be used in Groovy because the result of the logic element can ultimately end up in a 'double' which is currently serialized in JSON as a string. Therefore ensure you use 'BigDecimal' whenever dealing with a decimal point number and force the type when initially getting data back from a function call.

You can do this as follows:

```
num1.toBigDecimal()
```

As you can see in the Groovy documentation, there are situations where a mathematical operation between 2 different types can result in a third type. For example, a BigDecimal multiplied by a Float will end up in a Double!

Extract from the Groovy documentation:



5.5. Math operations

Although operators are covered later on, it's important to discuss the behavior of math operations and what their resulting types are.

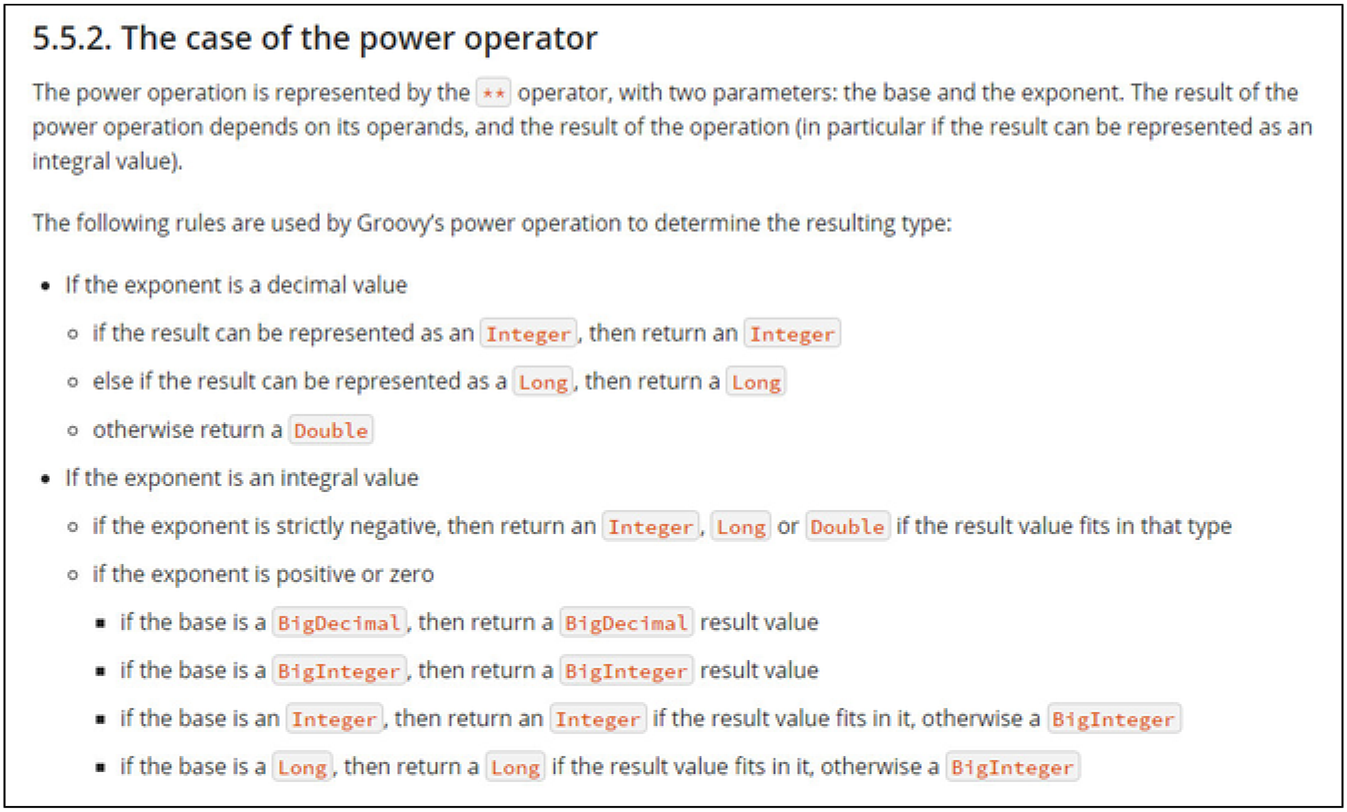
Division and power binary operations aside (covered below),

- binary operations between `byte`, `char`, `short` and `int` result in `int`
- binary operations involving `long` with `byte`, `char`, `short` and `int` result in `long`
- binary operations involving `BigInteger` and any other integral type result in `BigInteger`
- binary operations involving `BigDecimal` with `byte`, `char`, `short`, `int` and `BigInteger` result in `BigDecimal`
- binary operations between `float`, `double` and `BigDecimal` result in `double`
- binary operations between two `BigDecimal` result in `BigDecimal`

The following table summarizes those rules:

	byte	char	short	int	long	BigInteger	float	double	BigDecimal
byte	int	int	int	int	long	BigInteger	double	double	BigDecimal
char		int	int	int	long	BigInteger	double	double	BigDecimal
short			int	int	long	BigInteger	double	double	BigDecimal
int				int	long	BigInteger	double	double	BigDecimal
long					long	BigInteger	double	double	BigDecimal
BigInteger						BigInteger	double	double	BigDecimal
float							double	double	double
double								double	double
BigDecimal									BigDecimal

Also, there are situations in Groovy that will generate a 'double', for example the 'power of' operation:



5.5.2. The case of the power operator

The power operation is represented by the `**` operator, with two parameters: the base and the exponent. The result of the power operation depends on its operands, and the result of the operation (in particular if the result can be represented as an integral value).

The following rules are used by Groovy's power operation to determine the resulting type:

- If the exponent is a decimal value
  - if the result can be represented as an `Integer`, then return an `Integer`
  - else if the result can be represented as a `Long`, then return a `Long`
  - otherwise return a `Double`
- If the exponent is an integral value
  - if the exponent is strictly negative, then return an `Integer`, `Long` or `Double` if the result value fits in that type
  - if the exponent is positive or zero
    - if the base is a `BigDecimal`, then return a `BigDecimal` result value
    - if the base is a `BigInteger`, then return a `BigInteger` result value
    - if the base is an `Integer`, then return an `Integer` if the result value fits in it, otherwise a `BigInteger`
    - if the base is a `Long`, then return a `Long` if the result value fits in it, otherwise a `BigInteger`

So if you do such an operation, you should perform a conversion to 'BigDecimal' afterwards. A 'double' value which is serialized to text can have an unexpected impact; for example in a normal export to Excel where these values may not be recognized correctly depending on the user locale.

How Global Variables are Handled in Second Pass

Global variables are very useful to persist information either:

- between elements within the execution of a logic
- between executions of a logic for a set of items (i.e. PG, PL ...) if `api.retainGlobal = true` is called

This 'retainGlobal' also works for making use of global variables during executions of both 1st and 2nd pass (i.e. dirty pass). BUT given that the second pass is a whole new JST, the global variables are persisted between the 2 passes. The persistence is done in JSON.

So if you have a complex object (such as a 'Filter') and you store it in a global variable, this global variable can be retrieved from the global cache as a 'Filter' object as long as you are in the first pass. But during the second pass, if you retrieve it you will get the JSON representation of this object! Making use of this in your logic might cause errors.

Accessing the key of a map stored as a global variable in second pass is done using the syntax:

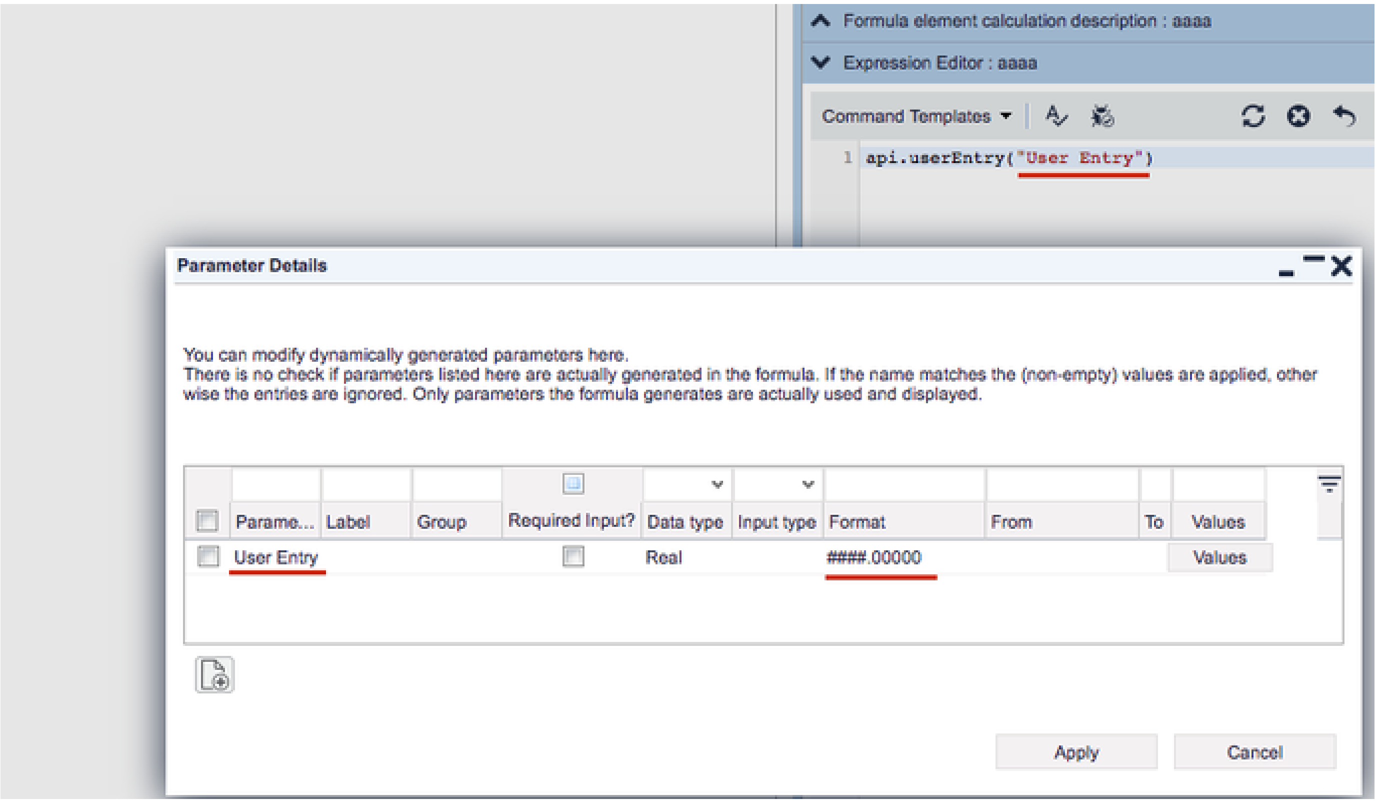
```
api.global.variable.map[(key)]
```

How to Set the Number of Displayed Decimals in User Input

Whenever there is a user input on the Line or Header level, the displayed value is automatically set to two decimals. Internally, the number is stored with higher precision and it is also used at this higher precision in the calculation logic.

You can change the user input format to match the Format Type configured in the logic.

1. In the calculation logic editing mode, click the 'Parameter Details' button.
2. Add a new line.
   a. Enter the 'Parameter Name', which must exactly match the parameter defined in the logic.
   b. In 'Format', specify the number of decimals that will be displayed in the user input field, e.g., ##.0000 for four decimals.
3. Click 'Apply' and then save the logic.



```
api.userEntry("User Entry")
```

You can modify dynamically generated parameters here.
There is no check if parameters listed here are actually generated in the formula. If the name matches the (non-empty) values are applied, other wise the entries are ignored. Only parameters the formula generates are actually used and displayed.

□	Parame...	Label	Group	Required Input?	Data type	Input type	Format	From	To	Values
□	User Entry			□	Real		####.00000			Values

You can also define and set the parameter using a Groovy script:

```
api.userEntry("User Entry")
def p = api.getParameter("User Entry")
p?.setConfigParameter("formatType", "###.00000")
```

How to Write Formulas

Written by:	Christian Tratz

Our formulas are used to calculate the results but at the same time also serve as configuration mechanism to drive certain UI elements, such as necessary input parameters.

Syntax Check Mode

To avoid the need to declare input parameters explicitly (they show up as you start using/expecting them in the formula), a mechanism called “syntax check mode” is used.

As the name implies, it not only determines inputs but also checks the formula for syntactical validity. This is e.g. the reason why a formula is executed in that mode once before saving it to prevent invalid formulas to be saved.

During such a syntax check execution, it is good to know:

- As formula inputs may actually be no real inputs as their data e.g. comes from previous calculation steps, the formula returns a default value for every input it encounters during this syntax check mode.
- This way the formula engine can in the end distinguish between “real” inputs (that it then needs to render to the user) and inputs that are fed by other formula elements (and do not trigger a UI element to be shown).

As you can see, determining inputs and executing formula are two calculation passes of the same “thing”. To not have negative performance impact, the syntax check passes are only done when required.

This is particularly relevant in e.g. a quote context. Here the input determination is only executed when the item is added to the quote; not every time the quote is recalculated.

If you require dynamic inputs that depend on other data or should change based on some other changes, you need to handle that either by header formulas (which can manipulate inputs) or use a configurator approach.

As a best practice, input parameters should be high up the logic, with no or as little as possible dependencies / conditional logic on them.

The following example shows how the `api.isSyntaxtCheck()` API call is used to determine which code does not need to be run in the parameter determination mode and should be called only in the real computation (like expensive DB calls).

```
def inputVal = api.userEntry()
if (api.isSyntaxCheck())  return;
def costs = api.find("PX", equal("name", "costs"))
```

How to Save Logic that Is Evaluated as Erroneous by Syntax Checker

There is a simple way to disable built-in syntax checker for the whole calculation, so you do not need to do it for each element separately. It works only for the save operation. While looking for configuration (selecting a logic), it will never abort any real calculations.

To set it up, create a new element, put it on the 1st or 2nd position and insert the following code in it:

This feature is not intended and will not work as a workaround for cases when the coding style is bad:

- If there are real issues (bugs) in your code, of course this does not help.
- If you have "null" references, it is a better practice to fix it by using "?:"

But, for example, when the code works fine in the Test Drive but it is not possible to save the logic, then you can consider disabling the checker.

How to Check Workflow Step Execution Status

If you want to find out about the execution status of a workflow step, you can use the following methods:

- isInitialized()
- isPreExecuted()
- isDenied()
- isApproved()
- isWithdrawn()

Example

```
for (step in workflowInfo.steps) {
 if (step.isPreExecuted()) {
  stageOwner = step.uniqueName
   }
}
```

How to Dynamically Generate Server Request from Dashboard

From the 3.4.3 Cuba Libre PR3 release, you can use the new `addDownloadButton` method to download outputs to Excel or PDF. See the Wizard Dashboard section and the API Documentation for details.

This article relates to Wizard Dashboard.

If you need to download an Excel output of the particular formula output, you need to call the backend API:

```
/pricefx/${api.currentPartitionName()}/formulamanager.executeformula/Cus
tPriceList_Report?output=xls&templateName=price_list
```

If you need to pass also the input parameters, you have to POST the following data:

```
{
  "data": {
    "Customer": "0000502097",
    "targetDate": "2017-03-27"
  }
}
```

For posting data we use the following JavaScript:

```
var req = new XMLHttpRequest();
req.open('POST',
'/pricefx/dana/formulamanager.executeformula/CustPriceList_Report?output
=xls&templateName=price_list', true);
req.responseType = 'blob';

req.onload = function (event) {
 var blob = req.response;
 var link=document.createElement('a');
 if (/*@cc_on!@*/false || !!document.documentMode)
{window.navigator.msSaveBlob(blob, filename);} //to make it work on IE11
 if (typeof InstallTrigger !== 'undefined')
{document.body.appendChild(link);} //to make it work on Firefox
 link.href=window.URL.createObjectURL(blob);
 link.download='CustPriceList_Report.xlsx';
 link.click();
};

req.send(JSON.stringify(
{
  "data": {
    "Customer": "0000502097",
    "targetDate": "2017-03-27"
  }
}
));
```

The final Groovy code is:

```
def callPayload = """{'data':{'Customer':'${customerId}',
'targetDate':'${new Date().format("yyyy-MM-dd")}'}}"""

controller.addHTML("""<button onclick="var req = new
XMLHttpRequest();req.open('POST',
'/pricefx/${api.currentPartitionName()}/formulamanager.executeformula/Cu
stPriceList_Report?output=xls&templateName=price_list',
true);req.responseType = 'blob';req.onload = function (event) {var blob
= req.response;var
link=document.createElement('a');link.href=window.URL.createObjectURL(bl
ob);link.download='CustPriceList_Report_${new
Date().getTime()}.xlsx';link.click();};req.send(JSON.stringify(${callPay
load}));">Download</button> """);
```

✓ If you need to serialize a filter so that you can send it to a 3rd party service, you can do the following:

```
def json = api.jsonEncode(pg.asFilter()) //pg is a variable representing a ProductGroup
//filter object serialized to JSON (instanceof String)
def map = api.jsonDecode(json)
//a Map parsed from JSON

def userFilter = api.filterFromMap(map)
//Filter constructed from a map
```

⚠ If the CSRF (Cross-Site Request Forgery) protection is enabled on the Pricefx instance you use, your request will be blocked. You can recognize it by getting an error like this in the log:

```
12:06:52.693 [qtp692782171-25364] WARN yourPartition/YourName
n.p.s.filter.CSRFBlockingFilter - Request
/se-primas/formulamanager.executeformula/YourLogic may have been forged;
returning SC_FORBIDDEN/COMMAND_NOT_AUTHORIZED
12:06:52.693 [qtp692782171-25364] INFO  yourPartition/YourName
n.p.s.filter.CSRFBlockingFilter - sessionToken:
1bo30ng8ur5dk1ia5ig3i7mj7f requestToken: null
```

What you need to do is:

How to Create an Interactive Embedded Dashboard

To create an interactive embedded dashboard:

1. Define a result matrix that will contain the data and will enable the user to trigger an event.
2. Define a portlet that will dynamically change depending on the triggered event.

Define Result Matrix

Create a result matrix and add a line that will trigger a client-side event when the user clicks on a row in the matrix. The value of the specified column for that row is passed on as an attribute. This will be the new input for the dependent portlet.

The event-triggering line will be similar to this one:

```
matrix.onRowSelection().triggerEvent(api.dashboardWideEvent("CustomerIdChanged-ED")).withColValueAsEventDataAttr("ProductID", "customerId")
```

When the user selects a row, a dashboard event called CustomerIdChanged-ED is triggered and the value of the ProductID column for that row is passed on as an attribute.

The element, which defines the result matrix, will be similar to this one:

```
def rawMatrix =
api.datamartLookup("ProdTX","ProductLabel","ProductID","InvoicePrice\$",
"NetMargin%-")
def resultMatrix = api.newMatrix("Margin Status","Product
ID","Label","Revenue","Net Margin %")
if(rawMatrix == null) return 0
def i=0
for(entry in rawMatrix.getEntries()) {
  def row =[:]
  row.put("Product ID",entry.get("ProductID"))
  row.put("Label",entry.get("ProductLabel"))
  row.put("Revenue",entry.get("InvoicePrice\$"))
  def margin = entry.get("NetMargin%")/100
  row.put("Net Margin
%",resultMatrix.styledCell(margin,"#0000CD",null,"bold"))
  if(margin <= 0) row.put("Margin
Status",resultMatrix.libraryImage("Traffic","3_red"))
  if(margin <0.1 && margin > 0) row.put("Margin
Status",resultMatrix.libraryImage("Traffic","2_yellow"))
  if(margin >=0.1 ) row.put("Margin
Status",resultMatrix.libraryImage("Traffic","1_green"))

  resultMatrix.addRow(row)
  i++
  if(i>100) break;
}
resultMatrix.setColumnFormat("Net Margin %",FieldFormatType.PERCENT)
resultMatrix.setColumnFormat("Revenue",FieldFormatType.MONEY_EUR)
resultMatrix.onRowSelection().triggerEvent(api.dashboardWideEvent("Produ
ctIDChanged-ED"))
 .withColValueAsEventDataAttr("ProductID", "ProductID")

return resultMatrix
```

Define Dynamic Portlet

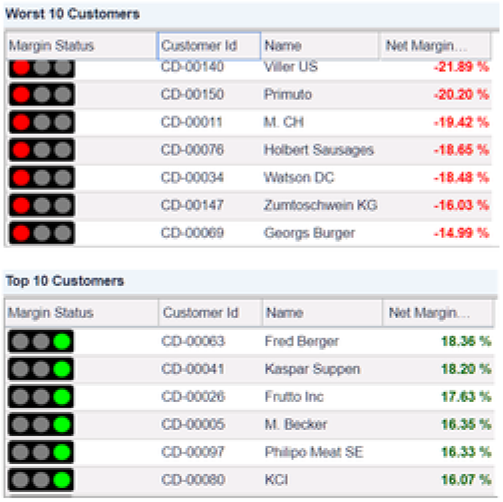
Create a dashboard that will be embedded in the master dashboard as a portlet. The content of this slave dashboard will dynamically change based on the received event.

The element defining the portlet will be similar to this one:

```
// Open/Show a dashboard named "Dashboard_Embedded"
return api.dashboard("Dashboard_Embedded")
// Here the user can assign values to the embedded
// dashboard inputs common to all three modes
.setParam("Product ID", api.input("Product ID"))
.setParam("Year", api.getElement("TimeEntry"))
// Show the dashboard embedded ...
.showEmbedded()
// ... and reevaluate it on "CustomerIdChanged-ED"
// Note: "dashboardWideEvent()" makes the event local to the containing
dashboard instance
.andRecalculateOn(api.dashboardWideEvent("ProductIDChanged-ED"))
// Pull the "Product ID" attribute out of the event payload and expose
it
// as the "Product ID" input to the embedded dashboard
.withEventDataAttr("Product ID").asParam("Product ID")
```

How to Use Traffic Lights in Data Tables

If you need to introduce a quick visual evaluation of your data, such as in a Top/Bottom Performers table, you can make use of the traffic lights (green, yellow, red) in the data tables. In the calculation logic you define the conditions corresponding to each color (see the example below).



Worst 10 Customers

Margin Status	Customer Id	Name	Net Margin...
🔴⚫⚫	CD-00140	Viller US	-21.89 %
🔴⚫⚫	CD-00150	Primuto	-20.20 %
🔴⚫⚫	CD-00011	M. CH	-19.42 %
🔴⚫⚫	CD-00076	Holbert Sausages	-18.65 %
🔴⚫⚫	CD-00034	Watson DC	-18.48 %
🔴⚫⚫	CD-00147	Zumtoschwein KG	-16.03 %
🔴⚫⚫	CD-00069	Georgs Burger	-14.99 %

Top 10 Customers

Margin Status	Customer Id	Name	Net Margin...
⚫⚫🟢	CD-00063	Fred Berger	18.36 %
⚫⚫🟢	CD-00041	Kaspar Suppen	18.20 %
⚫⚫🟢	CD-00026	Frutto Inc	17.63 %
⚫⚫🟢	CD-00005	M. Becker	16.35 %
⚫⚫🟢	CD-00097	Philipo Meat SE	16.33 %
⚫⚫🟢	CD-00080	KCI	16.07 %

The following images are ready to use:

```
resultMatrix.imageCell("/images/small_trafficlight_1_green.png")
resultMatrix.imageCell("/images/small_trafficlight_2_yellow.png")
resultMatrix.imageCell("/images/small_trafficlight_3_red.png")
```

```
def rawMatrix =
api.datamartLookup("CustTX","CustomerName","CustomerID","NetMargin%-")

def resultMatrix = api.newMatrix("Margin Status","Customer
Id","Name","Net Margin %")
if(rawMatrix == null) return 0

def i=0
for(entry in rawMatrix.getEntries()) {
  def row =[:]
  row.put("Customer
Id",resultMatrix.linkCell(entry.get("CustomerID"),"customersPage",entry.
get("CustomerID")))
  row.put("Name",entry.get("CustomerName"))
  def margin = entry.get("NetMargin%")/100
  row.put("Net Margin
%",resultMatrix.styledCell(margin,"#006300",null,"bold"))
  if(margin <= 0) row.put("Margin
Status",resultMatrix.imageCell("/images/small_trafficlight_3_red.png"))
  if(margin <0.1 && margin > 0) row.put("Margin
Status",resultMatrix.imageCell("/images/small_trafficlight_2_yellow.png"
))
  if(margin >=0.1 ) row.put("Margin
Status",resultMatrix.imageCell("/images/small_trafficlight_1_green.png")
)
  resultMatrix.addRow(row)
  i++
  if(i>10) break;
}
resultMatrix.setColumnFormat("Net Margin %",FieldFormatType.PERCENT)
return resultMatrix
```

Datamart Join Sample Code

```
// Get datamart context
def dmContext = api.getDatamartContext()
def productSku = api.product("sku")
// Create datamart query
def dmQueryInvoice =
dmContext.newQuery(dmContext.getDatamart("TX_Data"), false)
dmQueryInvoice.
  select("Sku").
  select("SalesOrder").
  select("UnitSellPriceExFreight").
  select("UnitSellingPrice").
  select("CostExSupplier").
  select("UnitCostExSupplier").
  select("Quantity").
  select("UnitCostAvg").
  select("UnitPriceList").
  select("Segmentation_FINAL").
  select("Quoted_Business").
  select("ListPrice").
  select("Sales").
  select("MarketingGroup").
  select("ProductCode").
  select("TransactionDate").
  where(Filter.isNull("SegmentationExclusions"),
        Filter.equal("Sku", productSku),
        Filter.equal("Active", "Yes"),
        Filter.equal("Team1","No"),
        Filter.notEqual("SalesWarehouse","95EX"),
        Filter.notEqual("ProductSpecial", "Yes"),
        Filter.isNull("Quoted_Business"),
        Filter.notIn("InvoiceType", "Credit", "Credit Note"),
        Filter.notEqual("Quantity", 0),
        Filter.notEqual("SalesCost", 0))

// Create datamart query
def dmQueryCreditNote =
dmContext.newQuery(dmContext.getDatamart("TX_Data"), false)
dmQueryCreditNote.
  select("Sku").
  select("SalesOrder").
  select("UnitSellPriceExFreight").
  select("UnitSellingPrice").
  select("CostExSupplier").
  select("UnitCostExSupplier").
  select("Quantity").
  select("UnitCostAvg").
  select("UnitPriceList").
  select("Segmentation_FINAL").
  select("Quoted_Business").
  select("ListPrice").
  select("Sales").
  select("MarketingGroup").
  select("ProductCode").
  select("TransactionDate").
  where(Filter.isNull("SegmentationExclusions"),
        Filter.equal("Sku", productSku),
        Filter.equal("Active", "Yes"),
        Filter.equal("Team1","No"),
        Filter.notEqual("SalesWarehouse","95EX"),
        Filter.notEqual("ProductSpecial", "Yes"),
        Filter.isNull("Quoted_Business"),
        Filter.notIn("InvoiceType", "Credit", "Credit Note"),
        Filter.notEqual("Quantity", 0),
        Filter.notEqual("SalesCost", 0))

def joinFields = [:]
joinFields.put ("Sku", "Sku")
joinFields.put ("SalesOrder", "SalesOrder")
def dmQueryCreditNoteAware = dmContext.newQuery(dmQueryInvoice,
dmQueryCreditNote, joinFields, false)
dmQueryCreditNoteAware.
  select("Sku_1", "Sku").
  select("SalesOrder_1", "SalesOrder").
  select("((UnitSellPriceExFreight_1 * Quantity_1) +
IfNull(UnitSellPriceExFreight_2, 0)) / Quantity_1" ,
"UnitSellPriceExFreight").
  select("((UnitSellingPrice_1 * Quantity_1) +
IfNull(UnitSellingPrice_2, 0)) / Quantity_1" , "UnitSellingPrice").
  select("CostExSupplier_1", "CostExSupplier").
  select("UnitCostExSupplier_1", "UnitCostExSupplier").
  select("Quantity_1", "Quantity").
  select("UnitCostAvg_1", "UnitCostAvg").
  select("UnitPriceList_1", "UnitPriceList").
  select("Segmentation_FINAL_1", "Segmentation_FINAL").
  select("Quoted_Business_1", "Quoted_Business").
  select("ListPrice_1", "ListPrice").
  select("Sales_1 + IfNull(Sales_2, 0)", "Sales").
  select("MarketingGroup_1", "MarketingGroup").
  select("ProductCode_1", "ProductCode").
  select("TransactionDate_1", "TransactionDate")

// Run query and save result to memory
def result = dmContext.executeQuery(dmQueryCreditNoteAware)
if (result) {
  def tableContext = api.getTableContext()
  tableContext.createTable("Transaction_NonQuoted", result)
}


if (api.isSyntaxCheck()) return

def tableContext = api.getTableContext()
def query = tableContext.newQuery("Transaction_NonQuoted")
query.
  select("UnitSellPriceExFreight")
query.where("TransactionDate >= '${api.getElement("TimeFrameStart")}'",
            "Quantity > 0",
                "UnitSellPriceExFreight > 0",
            "UnitSellingPrice < '${api.getElement("ListPrice")}'")
```

```
def result = tableContext.executeQuery(query)
result?.getColumnValues(0)
```

Logic Types Cheatsheet

#	Type	For Object	Usage	Where to Create	Where to Set Up	Input	Output
1	Calculation Flow		Scheduled Task	Administration > Configuration > All Modules > Calculation Flow Logics	Administration > Configuration > All Modules > Calculation Flows	Variable actionBuilder	
2	Configurator		Product Configuration	Administration > Calculation Logic > Generic Logic	api.configurator()		
3	Contract – Header Logic	CT	To find out classification of the contract or calculate summary results for the Contract.	Administration > Configuration > PromotionManager > Contract Header Logics		Variable "cProcessor"	
4	Contract – Term Type Filter Logic	CT	To show only filtered set of the TermTypes to the user when creating new lines in the Contract.	Administration > Configuration > PromotionManager > Contract Term Type Filter Logics		Variable "contract"	Filter for the Term Types. The logic can have multiple elements; for the resulting filter they are joined using the AND operator.
5	Contract – Entitlement Filter	CT	To limit access to the object detail screen.	Administration > Calculation Logic > Generic Logic	Administration > Configuration > PromotionManager > General Settings	Variable "contract"	When all elements return true, the user is allowed to view/edit details of the object.
6	Contract – Term Type	CTLI	To calculate one Contract line.	Administration > Calculation Logic > Contracts			
7	Customer – Detail Logic	C	To show additional information on the Customer Detail screen.	Administration > Configuration > PriceBuilder > Customer Detail Logics	Can be displayed on the "Customer Detail View" screen.		
8	Customer – Input Filter	Q, CT, RBA	In the Customer Input, to show only customers relevant for the logged-in user.	Administration > Calculation Logic > Generic Logic	api.customer(attributeName, customerId, defaultFromParent, filterFormulaName, filterFormulaParam)	Variable filterFormulaParam	A single element must return Filter (which will work on the Customer table).
9	Dashboard		To provide analysis overview, …	Administration > Calculation Logic > Generic Logic	Administration > Dashboards	For Embedded dashboards use api.dashboard()	Every visible element can return either Matrix or Chart (those will be displayed in the separate Portlets).
10	Dashboard Wizard		To provide user guidance – how to use the system step by step.	Administration > Calculation Logic > Generic Logic	Administration > Access Admin > User Admin	api.newController()	Every visible element returns a controller (those various HTMLs will be displayed in the separate sections of the wizard).
11	Feeder Logic	RR	To provide a list of RR to be allocated to transactions. Used for Rebate Allocation process.	Administration > Calculation Logic > Generic Logic	PriceAnalyzer > Data Manager > Data Loads		

12	Inline Details View	P, PX, C, CX, PL, PG, MPL	To create a pop-up window with line details over which you hover your mouse.	Administration > Calculation Logic > Generic Logic	Administration > Configuration > PriceBuilder > Inline Details Configuration		
13	Live Price Grid	PGLI	To calculate a PG line.	Administration > Calculation Logic > Generic Logic			
14	Matrix Logic	PL, PG	To provide values for second dimension of a Price List.	Administration > Calculation Logic > Generic Logic		api.product("sku")	A visible element must return a list of strings.
15	PriceAnalyzer Logic	DF, DS, DM		Administration > Calculation Logic > PriceAnalyzer	PriceAnalyzer > Data Manager > Data Loads		
16	PriceList	PLI		Administration > Calculation Logic > Generic Logic		api.userEntry()	Values of visible elements
17	Product Detail Logic	P	To show additional information (in the form of a portlet) on the Product Detail screen.	Administration > Configuration > PriceBuilder > Product Detail Logics	Can be displayed on the "Product Detail View" screen.		
18	Quote – Line Logic	QLI		Administration > Calculation Logic > Generic Logic			
19	Quote – Header Logic	Q		Administration > Configuration > QuoteConfigurator > Quote Header Logics		Variable "quoteProcessor"	
20	Quote – Product Filter Logic	Q	To limit the set of products the user can see when adding a new product to a quote.	Administration > Configuration > QuoteConfigurator > Quote Product Filter Logic		Variable "quote"	Filter for Products
21	Quote – Entitlement Filter	Q	To limit access to the object detail screen.	Administration > Calculation Logic > Generic Logic	Administration > Configuration > QuoteConfigurator > General Settings	Variable "quote"	When all elements return true, the user is allowed to view/edit details of the object.
22	Rebate Agreement –Entitlement Filter	RA	To limit access to the object detail screen.	Administration > Calculation Logic > Generic Logic	Administration > Configuration > RebateManager > General Settings	Variable "rebateagreement"	When all elements return true, the user is allowed to view/edit details of the object.
23	Rebate Agreement – Header	RA		Administration > Configuration > RebateManager > Rebate Agreement Header Logics		Variable "raProcessor"	
24	Rebate Agreement – Template Logic	RA		Administration > Configuration > RebateManager > Rebate Agreement Template Logics		variable "ratBuilder"	
25	Rebate Type	RA LI, RR		Administration > Calculation Logic > Rebates		Variables "calculationBase", "rebaterecords"	
26	Word Template Pre-processing Logic	Q, CT, RBA, PL, LPG, PR, RR	To prepare data for the Word template. Intended for export to PDF.	Administration > Calculation Logic > Generic Logic	Administration > Configuration > All Modules > Publishing Templates		
27	Workflow	Q, Contract, PGI, RA, RR, DCR, PL, PolicyRecord		Administration > Workflows > Workflow Logics		Variables "workflow", "submitter" and one of the following: quote, pricegriditem, rebateagreement, contract, dcr, pricelist, rebaterecord.	
28	Workflow Post Step	The same as Workflow		The same as Workflow. Switch to the post step logic via a dropdown.		The same as workflow + approvable (generic name for all approvables).	

29	Document Creation Workflow	Q, CT, RBA	Allows more users to cooperate on document creation.	Administration > Calculation Logic > Generic Logic (Nature: "Creation Workflow")	Administration > Configuration > All Modules > Workflows > Enable creation workflow for document type		
30	Workflow Delegation User Filter (available in 3.5 Salty Dog)		Allows administrators to narrow down the list of users available for workflow delegation.	Administration > Calculation Logic > Generic Logic			
31	Data Analyzer Table/Chart Series Formula	Rows of the series	When you need to enrich the data of the series in the chart (or table) in Data Analyzer. The results can be used in the chart definition in the control pane. Usage: - Additional columns for the series - Summary of the rows (SUM, AVG or any calculation) can be used e.g. for the auxiliary line	Administration > Calculation Logic > PriceAnalyzer	When you create a chart, and display the chart series table, it is at the top right.	- Query result - Row data, e.g. source Row[MaterialId]	Depends on the context of the logic element: - As a column of the series or - Summary on the right side next to the series data table

How to Get an Email Notification When a Background Job Fails

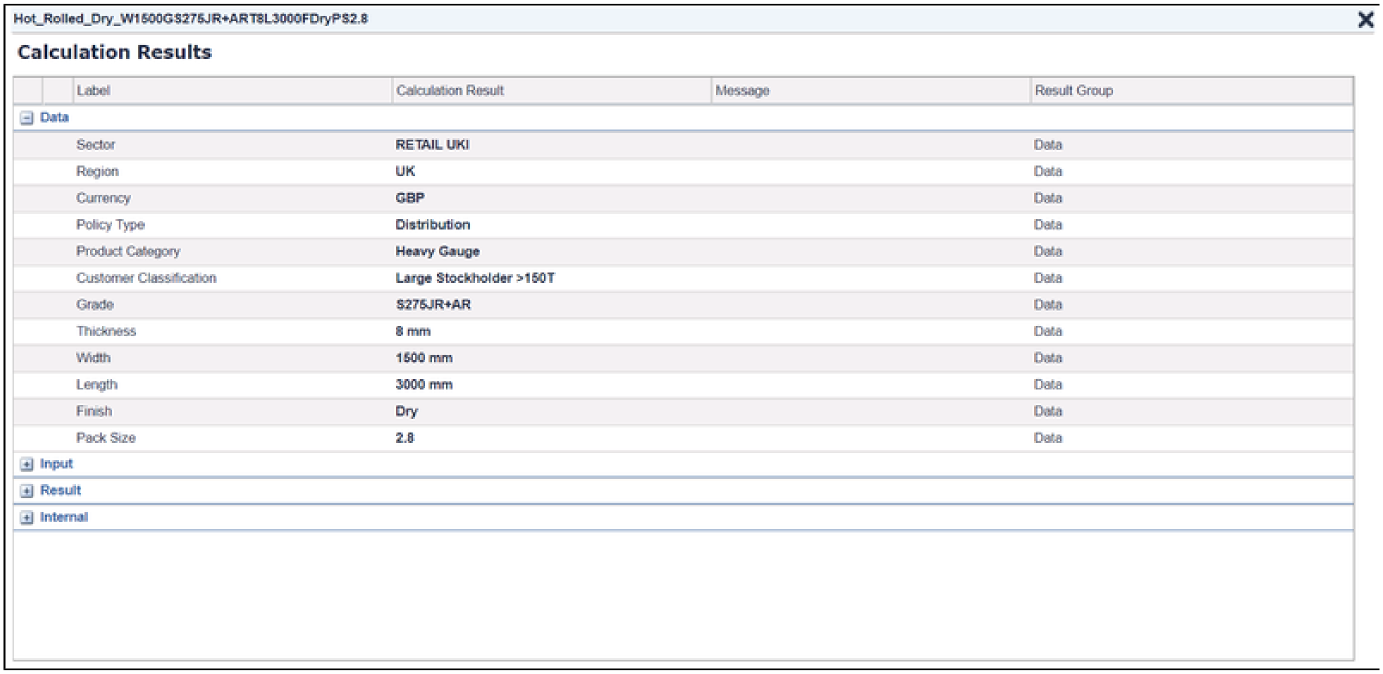
If you are running a large number of background jobs, for example Data Loads or Calculation Flows, it is useful to be notified of jobs, which failed due to an error or timeout.

By adding the advanced configuration property `sendEmailOnJSTFailure` and setting it to 'true', email notifications will be sent on every job failure. The emails will be sent to users, who have the 'Receive admin emails' option enabled in the User Admin section.

How to Group Calculation Results

In the calculation details popup window (e.g., in Price List details), it is possible to have the results by default grouped by the Result Group.

To take advantage of this feature, use the per-partition advanced configuration property `autoGroupCalculationResultsInDetailsView`. Set its value to "true".



Hot_Rolled_Dry_W1500GS275JR+ART8L3000FDryPS2.8

Calculation Results

Label	Calculation Result	Message	Result Group
⊟ Data			
Sector	RETAIL UKI		Data
Region	UK		Data
Currency	GBP		Data
Policy Type	Distribution		Data
Product Category	Heavy Gauge		Data
Customer Classification	Large Stockholder >150T		Data
Grade	S275JR+AR		Data
Thickness	8 mm		Data
Width	1500 mm		Data
Length	3000 mm		Data
Finish	Dry		Data
Pack Size	2.8		Data
⊞ Input			
⊞ Result			
⊞ Internal			

How to Set Up Failover Monitoring

You can enable a failover mechanism for the calculation flow processor by setting the ServerRole: "Calculation Flow Failover Processor" for the node(s) which should pick up processing if the master node (i.e. a node with "Calculation Flow Processor") fails.

Detailed scenario:

1. The master node becomes unavailable.
2. The other nodes notice it (through seeing an expired distributed lock during their heartbeat call) and increase a "failover counter". (This counter can be used for monitoring.)
   (The processing happens on the 2nd heartbeat. It does not start immediately to avoid collisions with master restarts/slower starts.)
3. On the next heartbeat, the first node picks the work and starts to process it.
4. Once the master is available again, the failover node notices it and increases the "recovery counter", and lets the master do its work again.

When there are two nodes and:

- Both are masters (i.e. have the "Calculation Flow Processor" role), then just a single one is allowed to process.
- Both are failover nodes and none is master: it works the same as described above, it just takes longer to decide which node will be processing (the failover node starts on 3rd heartbeat).

The counters mentioned above belong to the metrics counter and they are exported to monitoring. Based on them you can create your own alerts. "Tasks" in the metric class:

- `BackgroundCalculationFlowTaskWorker.failover` – Increased anytime a master vanishes and some other node gets ready for taking over.
- `BackgroundCalculationFlowTaskWorker.failoverProcessing` – Increased anytime a failover node does processing.
- `BackgroundCalculationFlowTaskWorker.recovery` – Increased anytime the master gets back and is expected that within the next heartbeat will take over the processing.

How to Assign a Unique Autogenerated ID to Price Parameters

> Available only in the classic UI (no Excel Client and LoadData support) from version 3.5 Salty Dog.

Some solutions may require price parameters to have a unique ID, which identifies the parameter and can be passed on or referenced. This ID also should be generated automatically and not be user-editable.

How to achieve this:

Add the autogenerateKeysInPP application property (in the advanced configuration) and set its value to true.
If any MLTV (MatrixLookupTableValue, i.e., Matrix PP value) has in its metadata set that a key is read-only, it will not be possible to insert a key in the UI, and the server will autogenerate a key with a (nearly) unique value that should be parsable even to double or long (64bit-signed) type.

This applies only to MLTV and DCR over MLTV (DataChangeRequest) add function.

How to Show customerId Value from customerGroupEntry Picker in Outputs

Let's assume you want to show the customerId value in the outputs.

If you used just `api.customerGroupEntry()`, then the result in UI would be shown as e.g. "CG['customerId' = '60838']".

The correct procedure to display the value '60838' is:

- Use `api.getElement("CustomerGroup").customerFieldValue`, depending on what is in the CustomerGroup element.
- First define the CustomerGroup element, visibility set to 'Never':

CustomerGroup

```
api.customerGroupEntry()
```

- Then in any other element doing the calculations, use a syntax check.
  Also, set the visibility of this element to 'QuoteConfigurator' at least.

SomeOtherElement

```
if (api.isSyntaxCheck()) return //as the first line
if (api.geElement("CustomerGroup").customerFieldName ==
"customerId") {
  return api.getElement("CustomerGroup").customerFieldValue
}
```

About Syntax Check

It is used in two use cases:

1. Anytime you save a formula, it checks whether the Groovy code is well formed and that the code does some additional mocking of dependencies of the current element (it usually tries to add a neutral element of the type but if it cannot infer the type, it will assume the BigDecimal type).
   So in this case, the syntax checker is unable to infer the return type of the expression `api.getElement("CG")` and will decide to put there a new `BigDecimal(1)`.

2. Usually any logic is run twice. In the first run, it just needs to see all calls to the methods which define input params (and so the isSyntaxCheck is set to true). Otherwise it is a normal logic run and this variable is set to false. The first run is really needed for the system to know what inputs the logic will generate.

> For more details on isSyntaxCheck see Miscellaneous Functions.

How to Avoid Slow Initial Rendering of Result Matrix?

Make sure you have no saved preferences that have the auto-fit feature set for the table. If you save such preferences, any initial load will be extremely slow, because the classic UI framework needs to do a lot of reflows.

How to Set Quote Owner

The following code shows how to set a quote owner. It can be useful e.g. in an integration scenario when Integration Manager handles the revisions. Without this code, the quote would be owned by Integration Manager and this information would be passed to 3rd party systems which is not ideal for audit or tracking purposes.

```
function void updateQuoteOwner(String uniqueName, int userId) {

 // fetch existing quote to update quote owner
 // in case of creation, use price operation
 Response resp = priceFxClient.getQuoteApi().fetch(uniqueName);

 if (resp != null && resp.getResponse() != null) {
     ArrayList quotes = (ArrayList) resp.getResponse().getData();
     Map<String, Object> quote = (Map<String, Object>) quotes.get(0);

     // set/update quote owner by his/her userId through ‘createdBy’
field
     quote.put("createdBy", userId);

     // save the update to Pfx
     priceFxClient.getQuoteApi().save(new SaveQuoteRequest().data(new
PriceQuoteRequestData().quote(quote)));
 }
}
```

How to Authenticate with Swagger

Written by:	Akash Duseja

This article describes how to authenticate with Swagger using Pricefx API.

Login

Before using majority of the Swagger functionality, the client application needs to authenticate with Pricefx. We use OAuth for authentication. Use the Base URL https://www.pricefx.eu/pricefx/<partition-name> to access the endpoints described below.

The login process uses the following endpoints:

Login: POST /token

Sample Body for /token POST request:

```
{
"username": "string",
"partition": "string",
"password": "string"
}
```

Sample Response for /token POST request:

```
{
"access-token": "string",
"token-type": "string",
"refresh-token": "string",
"expires-in": 0
}
```

Refresh token: POST /token/refresh

Sample Body for /token/refresh POST request:

```
{
"username": "string",
"partition": "string",
"password": "string"
}
```

Sample Response for /token/refresh POST request:

```
{
"access-token": "string",
"token-type": "string",
"refresh-token": "string",
"expires-in": 0
}
```

Logout: DELETE /token

Sample Response for /token DELETE request: No body

Response Code: 204

All 3 Authentication Requests should have the following headers:

```
Content-Type: application/json
Accept: application/json
Pricefx-Key: pfxKeyString
```

The value for the `pfxKeyString` in the headers can be provided by Pricefx upon request.

The login endpoint will return an access-token, an access-token-type and a refresh-token.

The access-token will have a defined non-auto-prolonging lifetime and can be renewed by the refresh-token and the refresh endpoint. The refresh-token does not have a lifetime, but is tied to the access-token.

The authentication of all subsequent requests is done by adding the access-token-type and access-token values separated by a space in the Authorization header as shown here:

```
Authorization: {access-token-type} {access-token}
```

Function Result Caching Examples

Have you ever wanted to run a function like api.find() inside a loop, and have it cache the result instead of re-executing the function for every iteration of the loop?

You can use the below helpers to cache the result either in shared cache, or in a local map.

Let's say you have a set of quote line items, and you need to fetch a list of products based on some filter for each and do something with the result set.

What if multiple line items have the same product filter on them, you don't need to hit the api each time. You just need a way to detect whether the filter is the same, and if so return the previously cached result.

In the example below, we use a local map to store the results of the api.find, using the filter as the key string. The next time that same filter is used, we will receive the previously returned result instead of hitting api.find() again.

Example Usage

```
def cache = [:]
lineItems.each { li ->

    def products = lib.cache.getOrSetMap(cache, "ProdSet-${li.filter}", filter, { filter ->
  return api.find("P", filter)
 })

 //do something with products
}
```

Below is the set of helper functions I have put into my Groovy library, in an element called "cache".

Note: Be responsible with the timeout. If you're trying to use a cache, but you have to increase the timeout to something huge, maybe you're using the wrong caching strategy.

lib.cache

```
/**
 * Cache objects in provided map, run closure to fill
 * @param map The map to store objects in
 * @param key Dictionary key to this object in map
 * @param args Array of inputs to give to closure
 * @param closure Function to run to populate cache of nothing exists there
 * @return Returns the result of the cache access, and mutates the cache map
 * EXAMPLE USAGE:
 * def lCache = [:]
 * def res = getOrSetShared(lCache, "SomeKey", [arg1, arg2, arg3], {
arg1, arg2, arg3 -> DoSomething() })
 */
def getOrSetMap(map, key, args, closure){
    def arr = args instanceof List ? args : [args]

    def res = map[key]
    if(!res){
        res = closure(*arr)
        map[key] = res
    }
    return res
}

/**
 * Same as getOrSetShared, but with an enforced Max TTL before refresh
is forced
 * Does not guarantee Minimum TTL, or removal from cache
 * Ironically uses cache to store TTL :)
 * @param maxTTL Max time in Seconds before object must renew in cache -
or be left for dead
 * @param key Shared cache key to this object
 * @param args Array of inputs to give to closure
 * @param closure Function to run to populate cache of nothing exists
there
 * @return Returns the result of the cache access
 * EXAMPLE USAGE: def res = getOrSetShared(3600, "SomeKey", [arg1, arg2,
arg3], { arg1, arg2, arg3 -> DoSomething() })
 */
def getOrSetShared(maxTTL, key, args, closure){
    def ttlFromNow = new Date().getTime()+(maxTTL*1000)//convert to MS
    def expiry = api.getSharedCache("TTL-${key}") // currently cached
expiry time, if any

    // if expiry is set, check it and expire if past
    if(expiry && (expiry as long)  >= ttlFromNow)
        api.removeSharedCache(key)
    else // if expiry not set, set it
        api.setSharedCache("TTL-${key}", api.jsonEncode(ttlFromNow))

    return getOrSet(key, args, closure)
}

/**
 * Cache objects in shared cache, run closure to fill
 * @param key Dictionary key to this object in cache
 * @param args Array of inputs to give to closure
 * @param closure Function to run to populate cache of nothing exists
```

```
there
 * @return Returns the result of the cache access
 * EXAMPLE USAGE: def res = getOrSetShared("SomeKey", [arg1, arg2,
arg3], { arg1, arg2, arg3 -> DoSomething() })
 */
def getOrSetShared(key, args, closure){
    def arr = args instanceof List ? args : [args]

    def res = tryDecodeJSON(api.getSharedCache(key))
    if(!res){
        res = closure(*arr)
        api.setSharedCache(key, api.jsonEncode(res))
    }
    return res
}

/**
 * An unfortunate hack I couldn't make more elegant...
 * Takes in a string of json and parses either a map or a list from it
depending on first character
 * If first character not '[' or '{', will just return what it was given
 * @param jsonString The string to attempt to parse from
 * @return List[], Map[:], or Object.
 */
def tryDecodeJSON(jsonString){
    if(jsonString instanceof String){
        if(jsonString.startsWith("["))
            return api.jsonDecodeList(jsonString)
        else if(jsonString.startsWith("{"))
            return api.jsonDecode(jsonString)
    }
    else return jsonString
}
```

Questions and Answers

Search the existing topics first:

Registered users can also post a new question here. Just click the **Plus** icon on the left.

Topic	Author	
How can I filter on a quote type?	Jana Volencova (8 days ago)	1
How can I set format for column in matrix input?	Jana Volencova (12 days ago)	3
Why is SUM not working on Datamart query	Aaron Freeman (15 days ago),...	1
How to Format Datamart Query Results	Jana Volencova (15 days ago),...	1
How to enable HighCharts regression lines and Data tab?	Aaron Freeman (19 days ago),...	1
How do I set direct approval/denial via email without intermediate step?	Jana Volencova (22 days ago)	1
Why are some cells in my Price List grayed out?	Lauire McLeod (41 days ago)	1
When adding attachment in QC on partition with thousands of user groups, I get error	Jana Volencova (57 days ago)	1
How to plot data on Z axis	Aaron Freeman (84 days ago),...	2
How do I tell which fixes from the change log made it into a specific build?	Christian Tratz (988 days ago),...	1
How can I determine which products are visible to a user?	Jan Panousek (782 days ago),...	4
PA Related Questions on Pricefx Backend JSON API	Jana Volencova (82 days ago)	1
How to include attributes with NULL values in Custom Events (omitNullFields = false)?	Jana Volencova (99 days ago)	1
Why there are different response times for the same call when using a different account?	Christian Tratz (113 days ago),...	1
Calculation Flows don't run	Jana Volencova (130 days ago)	1
How do I hide External Reference field in Contract Header Info?	Jana Volencova (184 days ago)	1
How do I get started with PromotionManager?	Henrik Alén (Bynfo, Partner) (1003	3

	days ago),...	
When to use Product Extension and when Price Parameter?	Pavel Matyasko (1273 days ago),...	2
What is the fastest way to get the first row of a DataMartQueryResult as map?	Sean Missingham (246 days ago),...	1
How can I set a column of InputMatrix to be read-only?	Jana Volencova (247 days ago)	1
How can I make Price List outputs mandatory?	Jan Panousek (523 days ago),...	1
Why is it not possible to query Integer column?	Lukasz Wiktor (278 days ago),...	1
Why can't I query a Boolean column?	Lukasz Wiktor (278 days ago),...	1
Is it possible for result matrix to specify labels different than keys?	Lukasz Wiktor (278 days ago),...	1
How do I get data from PriceAnalyzer in the logic?	Lukasz Wiktor (278 days ago),...	1
How do I use Datamart IF condition in SQL Expression?	Lukasz Wiktor (278 days ago),...	1
Can I query a Data Source directly?	Lukasz Wiktor (296 days ago),...	1
How to invoke a library function from another library function (defined in another element)?	Lukasz Wiktor (296 days ago),...	1
How can I add filter conditions as OR?	Jana Volencova (320 days ago)	1
Is there a way to read rebate header input from RR logic?	Jana Volencova (369 days ago)	1
Can I send notification email to quote creator instead of submitter?	Jan Panousek (421 days ago),...	1
Is it possible to filter a field based on its length?	Jana Volencova (370 days ago)	1
Is it possible to filter the DS text attribute by length?	Jana Volencova (370 days ago)	1
Which objects support deep links?	Jana Volencova (449 days ago)	1
What resultType parameter to use to display chart created with chart builder API?	Jana Volencova (467 days ago)	1
Why the contract IDs are not sequential?	Jana Volencova (523 days ago)	1
What to do if Price Records are not showing?	Jana Volencova (527 days ago)	1
Can I do Group By for results of api.find()?	Pavel Matyasko (553 days ago),...	1
Can header logic modify the expiry date on quote copy?	Jana Volencova (553 days ago)	1
What is Syntax Check good for?	Pavel Matyasko (561 days ago),...	2
How to set width of table columns placed in dashboard portlet?	Pavel Matyasko (562 days ago),...	1

How to change colors of PA charts?	Pavel Matyasko (562 days ago),...	1
Is it possible to change precision for Result Price on LPG?	Jana Volencova (565 days ago)	1
Life cycle of an InMemory table in a PA Calculation	Jana Volencova (599 days ago),...	1
Is there a separate transaction for each DataFeed row?	Jana Volencova (599 days ago)	1
What type of Object can be emitted by DataFeeder?	Jana Volencova (599 days ago)	1
Row count limit for InMemory tables and handling large datasets	Jana Volencova (599 days ago)	3
Can I have two business calendars generated into DM?	Jan Panousek (641 days ago)	1
How can I add warnings to folder level outputs that will display in calculation results?	Jana Volencova (657 days ago)	1
How do I specify the table to use for the Matrix Lookup input type?	Jan Panousek (664 days ago)	1
How to pass results calculated in the quote header logic to the line items?	Pavel Matyasko (670 days ago),...	1
Why is a quote line logic called twice when I add a new line?	Pavel Matyasko (670 days ago),...	1
Is it possible to add data to a product extension from within a workflow logic?	Jan Panousek (685 days ago)	1
How can I format calculation results in email notifications?	Jan Panousek (691 days ago),...	1
How do I get the whole row object within an element of a calculation logic?	Jana Volencova (694 days ago)	5
Should I access query results by field names or aliases?	Jana Volencova (727 days ago)	
How can I implement a relay state that defines a specific locale?	Jan Panousek (728 days ago)	1
How to add a string to a workflow email	Jochen Schmidt (890 days ago),...	1
How can I read data from an existing data source?	Jana Volencova (750 days ago)	2
How do I find a specific value in a line item?	Jana Volencova (900 days ago)	5
How do I get inputs/outputs of quote line items?	Jana Volencova (900 days ago)	1
Does the Competiton Data table have a limited number of columns?	Jana Volencova (918 days ago)	1
How to use shared cache	Jochen Schmidt (933 days ago),...	1
I can't see prices/dates in the Datamart	Jan Panousek (944 days ago),...	1
How to delete Price Lists from logic?	Daniel Knittel (1033 days ago),...	1
How to create my own currency?	Adrian Reich (935 days ago),...	1

Why in Dashboard view the date filter (DD/MM/YYYY) only accepts day numbers < 20?	Sampsa (967 days ago),...	1
How come when I api.delete something that I can still api.find it?	Christian Tratz (985 days ago),...	5
How can I manipulate with datetimes?	Jochen Schmidt (987 days ago),...	3
Why is the manualResultPrice not changed until the next calculation?	Jana Volencova (986 days ago),...	1
If a field in the LPG is manually changed, how to trigger a line recalculation?	Jochen Schmidt (988 days ago),...	2
How do I get today's date in a Groovy script?	Jana Volencova (994 days ago),...	3
How to add a manual editable attribute in a LPG, based on restricted fields (like the "option()" in Java for quotes?	Jochen Schmidt (995 days ago),...	3
How to "restart" aborted calculation of PL	Radek Feber (1027 days ago),...	1
When working with the Excel Client, what is the worksheet size limit?	Jana Volencova (1029 days ago),...	1
Do you have any sample Groovy code how to update a PriceGridItem record from CFS?	Jana Volencova (1035 days ago),...	1
Where can I call api.sendEmail?	Jana Volencova (1036 days ago),...	2
Is there any limit on the size of an attachment to a contract?	Jana Volencova (1044 days ago),...	1
Is there a way to make fields in PX or PP tables not editable?	Jana Volencova (1047 days ago),...	1
How can I fill anything into the Description column inside the Calculation results section?	Jana Volencova (1054 days ago),...	1
Which elements must be selected in LPG as output elements?	Michal Smisek (1062 days ago),...	1
How do I throw an exception?	Jana Volencova (1064 days ago),...	5
Is PFX compatible with a tablet or mobile devices?	DisabledAccount#1029 (1065 days ago),...	2
Can I use the Groovy library in PA?	Jana Volencova (1078 days ago),...	1
How to debug strange application behavior	Jana Volencova (1078 days ago),...	3
How can I avoid label resetting within sync DSs?	Daria Maksimova (1176 days ago),...	1
How to find Groovy functions in Confluence?	Pavel Matyasko (1239 days ago),...	1
How can I export and import all data and metadata from a partition?	Pavel Matyasko (1272 days ago),...	7
How can I set a warning on a price list item in Groovy	Pavel Matyasko (1267 days ago),...	2
How I can resolve 'too many new instances created: 503" error?	Milan Boruvka (1131 days ago),...	6
How can I get better search results here?	Jan Panousek (944 days ago),...	1

How can I filter on a quote type?

The following example does not work, it says there is no "quoteType" on the Quote class:

```
api.find("Q", Filter.equal("quoteType", "Configuration"))
```

How can I set format for column in matrix input?

How can I set the format (e.g. money or numeric) for a column in a matrix input?

How to enable HighCharts regression lines and Data tab?

I'm building a HighChart and using the code below (found in highcharts.com documentation) to enable regression lines and show a Data tab with the chart. However, the regression line and Data tab are not rendering in PFX. Can you verify that the correct modules needed for these functions are available in PFX?

Regression line code

```
regression: true,
regressionSettings: [
                type: 'logarithmic',
                color: 'rgba(0,0,0,1)',
                name: 'Regression'
],
```

Show Data tab code

```
.showDataTab()
```

Thank you!

How do I set direct approval/denial via email without intermediate step?

Is there a way **not** to render the pricegriditem_approvalRequiredForm form after clicking the approve link from a workflow notification email? The customer does not want to click twice.

When adding attachment in QC on partition with thousands of user groups, I get error

When I try to add an attachment in QuoteConfigurator on a partition with thousands of user groups, I get an error.

How to plot data on Z axis

How to plot data on Z axis when using a Bar & Line chart? I've enabled the Z axis but this seems to only be a scale to the right. For example: How do I specify that Series 4 data will be plotted on the Z axis?

How to include attributes with NULL values in Custom Events (omitNullFields = false)?

I created a map containing PX data that could include values or null. If I send this map using api.customEvent(), entries with null value are not included. Is this the expected behavior? What can we do to get the missing "key : null" pairs?

Why there are different response times for the same call when using a different account?

When making direct API calls, the response times for the same call using different users can be differing by 500ms. Access for both accounts is the same. Why is this?

Calculation Flows don't run

We have a problem with Calculation Flows on our partition, it seems that they just don't run, neither when when you click "Run now" nor when the scheduled time comes. Any ideas why?

How do I hide External Reference field in Contract Header Info?

How can I set a column of InputMatrix to be read-only?

What is the fastest way to get the first row of a DataMartQueryResult as map?

What is the fastest way to get the first row of a DataMartQueryResult as map?

Best I've found is this, but seems inefficient to convert between so many types first:

```
dmctx.executeQuery(q).getData().toResultMatrix().getEntries().getAt(0)
```

How can I make Price List outputs mandatory?

Sometimes, it might be necessary not to allow users to deselect important price list outputs. Missing outputs can cause integration issues, when, for example, importing approved price lists to SAP.

Why is it not possible to query Integer column?

Why can't I query a column "ColumnNumber" if it is set as Integer in Data Source? I need to agregate by the first 4 columns, so I set the rollup to true. The first 4 columns are keys in the Data Source.

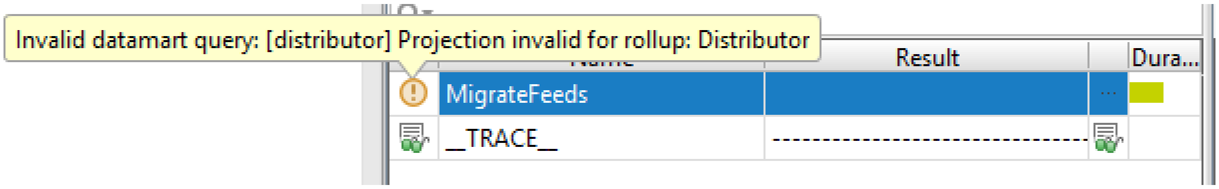
"ColumnNumber" is filled in the Data Source.

Why can't I query a Boolean column?

Why is it not possible to query a column of the Boolean type?

`Distributor` is a Boolean and when I run the query then it returns `null` and the following warning:

> *Invalid datamart query: [distributor] Projection invalid for rollup: Distributor*



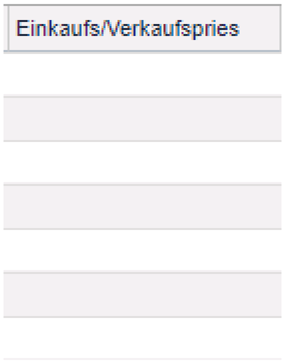
Invalid datamart query: [distributor] Projection invalid for rollup: Distributor

Name	Result		Dura...
MigrateFeeds		...	
__TRACE__	----------------------------------		

Is there any workaround for that?

Is it possible for result matrix to specify labels different than keys?

I need headers with special characters in the result matrix. When I add for example a slash / then no data appear in this column.

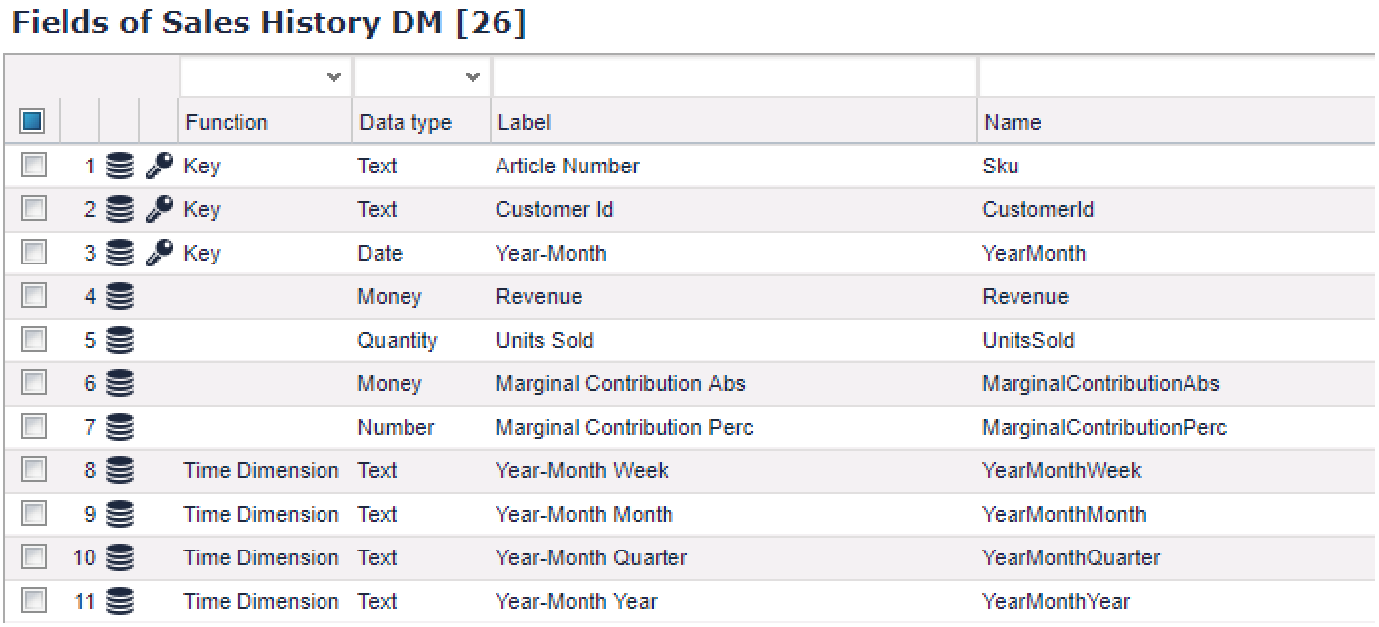


Einkaufs/Verkaufspries

How do I get data from PriceAnalyzer in the logic?

I have a Datamart named "SalesHistoryDM" with the following columns:

- Sku
- CustomerId
- YearMonth (Time Dimension)
- Revenue
- UnitsSold
- MarginalContributionAbs
- MarginalContributionPerc



Fields of Sales History DM [26]

□		Function	Data type	Label	Name
□	1	Key	Text	Article Number	Sku
□	2	Key	Text	Customer Id	CustomerId
□	3	Key	Date	Year-Month	YearMonth
□	4		Money	Revenue	Revenue
□	5		Quantity	Units Sold	UnitsSold
□	6		Money	Marginal Contribution Abs	MarginalContributionAbs
□	7		Number	Marginal Contribution Perc	MarginalContributionPerc
□	8	Time Dimension	Text	Year-Month Week	YearMonthWeek
□	9	Time Dimension	Text	Year-Month Month	YearMonthMonth
□	10	Time Dimension	Text	Year-Month Quarter	YearMonthQuarter
□	11	Time Dimension	Text	Year-Month Year	YearMonthYear

I need to aggregate the Datamart in the logic for a specific product. I need to sum Revenue, UnitsSold, MarginalContribution and count how many customers bought the product per year.

How do I use Datamart IF condition in SQL Expression?

I would like to calculate a field according to a value of another field in the Datamart but am unable to do so. What is the correct expression?

I tried:

```
IF(ColumnNumber=1,GrossSales,"")
CASE ColumnNumber WHEN 1 THEN GrossSales END
```

How to invoke a library function from another library function (defined in another element)?

Let's say I have `Element1` including function `f1`:

```
def f1() {
  return 1
}
```

I tried to invoke `f1` from `f2` defined in `Element2`. Both elements are in `__LIBRARY__` logic.

```
def f2() {
  return lib.Element1.f1() + 1
}
```

and I got an error "No such property: lib".

I also tried:

```
def f2() {
  return Element1.f1() + 1
}
```

which works in a regular logic but in the library I'm getting error "No such property: Element1".

I'm calling `f2` in this way:

```
lib.Element2.f2()
```

Element1 is defined before Element2.

What is wrong? Is there any way to call `f1` from `f2`?

Can I query a Data Source directly?

Does anyone know of a way to query data straight out of a data source from a calculation logic instead of setting up a Datamart + Data Load for it?

I found that I can use the search API to find the DS, but it contains no actual data, only the structure:

```
api.find("DMDS", Filter.equal("uniqueName", "DSNAME"))
```

How can I add filter conditions as OR?

In statements like `api.find("PGI", startRow, Filter.in("approvalState", "APPROVED", "DENIED", "AUTO_APPROVED"))`, the filters conditions are all cumulative and added as AND. Is there a provision to add them as OR?

Is there a way to read rebate header input from RR logic?

Is there a way to read the rebate header input from the RR logic? I added a hidden input in the header logic using raProcessor.addOrUpdateInput() to each line item, but while processing RRs, it is always empty. Any hints?

Is it possible to filter a field based on its length?

I want to filter SKUs that are less then 11 characters long. I need to do it in the UI.

Is it possible to filter the DS text attribute by length?

I need to filter the entries of the Attribute "MotorBM" in the Datasource "DS_Vehicle" where the length is greater than 6 characters. The attribute type is "Text", so I can't just filter entries greater than 999999. Is there any query or hidden filter function or some kind of workaround I can use?

Which objects support deep links?

I tried to create a deep link for a particular product extension and it just goes to the PX page and the app freezes. Is this a bug or is deep linking to a PX not supported?

What resultType parameter to use to display chart created with chart builder API?

What will be the value of the "resultType" parameter to display a chart (i.e. waterfall, scatter, time series created using the chart builder API) from the quote header? For example, to display a data table in the quote header, the "resultType" parameter value is "MATRIX", for gauge it is "GAUGE". What will be the value for charts? The output field is created using the addOrUpdateOutput method.

```
qp.addOrUpdateOutput(folderId, ["resultName": "Deal Performance",
"resultLabel": "Deal Performance",
"resultType": ??
"result" : waterfallChart])
```

Can I send notification email to quote creator instead of submitter?

Workflow notification emails are sent to the quote submitter when a quote is approved or denied. Is it possible to send the email to the quote creator instead?

How is the data for calendar DS generated?

What is the logic behind DataLoad that generates data for the calendar data source? Specifically:

1. How far to the future does it generate data?
2. On every project, should we schedule this job to run automatically after some time to avoid situation where we "run out" of the data there?

Why the contract IDs are not sequential?

On our production partition, we have so far created only 2 contracts, but the IDs are C-17 and C-23. Why there are gaps in the numbers? Our auditors might ask about the missing numbers.

What to do if Price Records are not showing?

Can I do Group By for results of api.find()?

When searching the product extensions (or product master), it would be sometimes useful to be able to calculate SUM or AVG of certain columns. I.e. we need to make a Group By for api.find(). Is that possible?

Can header logic modify the expiry date on quote copy?

I have a quote with the expiry date already in the past and when I create a copy of the quote, the date changes to "today". The effective date is not copied from the original quote and in the copy it is also set to "today". So I end up with a quote which has both effective and expiry dates set to "today" which is not ideal.

Is it possible to have the header logic to set the dates like this: expiry = today + x days?

What is Syntax Check good for?

When I save the logic, sometimes I get a Syntax Check error and the logic is not saved. What is this check good for?

How to set width of table columns placed in dashboard portlet?

How exactly can we set (via a calculation logic) the width of columns of a result matrix table placed in a dashboard portlet?

How to change colors of PA charts?

How can we change colors of the default charts found in Price Analyzer > Data Analyzer > Bubble Chart (or others)?

Is it possible to change precision for Result Price on LPG?

Is it possible to change precision for Result Price on LPG? I.e. from 2 decimals to 3, 4,..?

Row count limit for InMemory tables and handling large datasets

I encountered the 100.000 row count limit for the InMemory table. Is this limit for each InMemory table created in the current context or is it the sum of all rows of all the InMemory tables created in the current context? I created two InMemory tables, each with less then 100.000 rows but I got an error of the 100.000 row count limit. In case, the row count limit is sum of all InMemory tables created in the current context, how can we handle a huge dataset?

Life cycle of an InMemory table in a PA Calculation

When an InMemory table is created by a PA Calculation logic, is it preserved across row evaluations?
Is the behaviour different when a feeder is used?

What type of Object can be emitted by DataFeeder?

Is there is a plan to allow DataFeeder to emit any type of Object which can be further be used to load limited set to rows sequentially to the Datamart?

Is there a separate transaction for each DataFeed row?

Do the database transaction also start and end with the Calculation Logic (i.e. separate transaction for each DataFeed row)?

PA Related Questions on Pricefx Backend JSON API

The following questions relate to a specific subset of the PFX JSON API – the setup of PriceAnalyzer Datamarts (and underlying Data Feeds and Data Sources).

The general rule is that the Pricefx UI uses the very same API. So if you are unsure how things would work, open the UI, turn on e.g. Firebug or any other console that shows the browser requests and take a look while performing the necessary steps in the UI.

- Adding New Data
- Defining Field Names and Data Types of New Data Table
- Format of Data in Request Body
- Getting Data from DF to DM
- Adding Charts
- Using Groovy

Adding New Data

Question: I’ve found the /add/{type code} API in the general datasource section. Do you have a list of type codes? I’ve found the Data Sources object using the /fetch API but the type code isn’t obvious. Is it “DMDS”? I also found “typedId” from /datamart.getcharts and an example “typed ID” in the /datamart section, but the example “101.DS” seems a bit cryptic. Is 101 supposed to be a unique name of this Data Source? Do I first need to check that 101 is not already being used?

Answer:

The general datasource commands may not be the right ones. Anything related to our analytics module primarily goes through the “datamart” ones.

To add a new Data Feed (DF), Data Source (DS), or Datamart (DM) we use the datamart.newfc/<type> command, where <type> can be DMDF, DMDS, or DM. The request JSON should feature a "label" (free text) and a "uniqueName" (no special charts other than _). The response will feature the "typedId".

Generally the IDs are auto assigned. And command that generates a new object does not need that info in the request obviously. But it may be in turn required for data retrieval.

```
== /fetch Response ==
{
      "massEditable" : false,
      "bulkLoadable" : false,
      "name" : "DMDATASOURCE",
      "codeString" : "DMDS",
      "className" : "DMDataSource"
    }

== / datamart.getcharts Response ==
{
      "version" : 6,
      "typedId" : "123.SC",
      "uniqueName" : "Pricemanager__1347284372345",
      "label" : "Revenue per Year",
      "createdByName" : "Pricemanager",
      "chartDefinition" : …
```

Defining Field Names and Data Types of New Data Table

Question: How do I define the field names and data types of the new data table? I'm expecting a JSON version of SQL where I can define field names, data types, keys, uniqueness, default values, etc. What data types are available? Do I need to create both a Data Feed table (using all strings) and a Data Source table (with all the data types)?

Answer:

This is not quite covered in the Swagger Doc, as we've been considering this as a UI only task up to now. It is best to look at the browser requests. Look for datamart.updatefc. The backend will create a matching DF for any DS you create or update.

Format of Data in Request Body

Question: Do I use the /loaddata API to load string-only data into a data feed. Can you give me an example of what the data looks like in the request body.

Answer:

Be aware that there are two loaddata "flavours". One for analytical data, one for "normal" data. The request format for both is very similar.

The data types in the request can (and should) be the three types JSON supports (string, number, boolean). For other types (e.g. dates) we use a standard string representation of it.

Loaddata consists of "header" and "data" sections. Both are lists, so that the field names aren't repeated every time.

Some crude example:

```
“header”:[“ProductId”,”ProductName”],
“data”: [
[“001,”ABC”],
[“002,”DEF”],
…
]
```

You could also use the file upload with CSV (separator can be configured) or Excel.

Getting Data from DF to DM

Question: What is the correct sequence of APIs to get the data from the Data Feed to the Datamart? Do I always have to first load directly into a Data Feed, or can I skip that step and go to a Data Source? It looks like I maybe have to do back-to-back /datamart.util/flush/{DS typedId} then /datamart.util/flush/{DM typedId}?

Answer:

Also here we suggest that you take a look into how it works from the UI.

Via the file upload you can go straight to the DS, but error reporting will be really scarce. So yes, usually you'd flush the DS and then the DM.

Adding Charts

Question: How do I add charts? I think I use the same /add/{type code} API from above for the data table, but use a chart type code instead. Is it so?

Answer:

Charts are a bit tricky. The reason is that they do not really exist on the backend. What you see are “SavedCharts” objects. But these are just data blobs for persistence that are then used in the client to restore them and render them.

The answer here depends on what you would like to achieve.

- If you want to show charts in your UI, you may just query for the raw data.
- If you want to “inject” a set of predefined SavedCharts into the blank partition, then you should more or less generate them manually once and then treat that data as a blob (i.e. insert it as is).

Using Groovy

Question: Is your Groovy implementation just designed for writing scripts within Pricefx itself, or should I be thinking about using Groovy as an easy way to integrate with Pricefx from the outside?

Answer: Groovy is meant to live inside, not as an external API.

Can I have two business calendars generated into DM?

I have one definition in the Data Source and then it generates fine into the Data Mart. But I need two different business calendars generated into DM (sales and rebate dates).

How can I add warnings to folder level outputs that will display in calculation results?

How do I specify the table to use for the Matrix Lookup input type?

How to pass results calculated in the quote header logic to the line items?

Sometimes you need to evaluate the whole quote and then pass that result to the line items.

Some of the use cases:

- You need to evaluate the "type" of the quote – if the products are from the same category or from mixed categories. Depending on that you will make different calculations for the line items.
- You want to evaluate the mix of the line items and in certain cases add a new line (via the header logic) to the list of products on the quote. But you want to avoid the situation that this addition will happen again the next time you press "Calculate Quote". So you need to set some "flag" on the line item which causes the new line being added.

In both these examples we need to pass some information from the header logic to the line item.

Why is a quote line logic called twice when I add a new line?

When you have a quote and add a new line item, either via a button in the user interface or via a quote header calculation logic, the quote line logic is executed twice. The first time with `api.currentItem()` returning null and the second time with `api.currentItem()` returning the whole line. Why is that so?

Is it possible to add data to a product extension from within a workflow logic?

How can I format calculation results in email notifications?

Hi team, again a question for Workflow Notification Template.

I have a calculation result Element about new promotion Price. Result is: 4.2095 (a number with 4 decimal places). When I add this element to the Workflow Template, it is printed as 4.21 (with 2 decimal places only). How can I keep the original format?

Should I access query results by field names or aliases?

Hello every,

I've experienced that the query result (i.e. List of Maps) was able to be accessed by only field names. But when I run again the same codes, it returns me only aliases.

I wonder if this change is done on purpose, so that I will switch my codes to using aliases instead.

Below is example:

1. I have this query against T1 and T2 which are tables created (and cached) from querying DataMart:

def commonBusiness = tblCtx.executeQuery("SELECT T1.CustomerId as T1CustomerId, T1.Sku as T1Sku, T1.IP as T1IP, T1.V as T1V, T1.IPpu as T1IPpu,T2.CustomerId as T2CustomerId, T2.Sku as T2Sku, T2.IP as T2IP, T2.V as T2V, T2.IPpu as T2IPpu FROM T1 INNER JOIN T2 ON T1.Sku=T2.Sku AND T1.CustomerId=T2.CustomerId")

2. Then I have other codes to iterate above query results:

```
 cb.each {
   api.trace("row",null,it)
   def t2IPpu = it["T2.IPpu"]? it["T2.IPpu"] : 0
   def t1IPpu = it["T1.IPpu"]? it["T1.IPpu"] : 0
   if (it["T2.V"])
     effect += (t2IPpu - t1IPpu) * it["T2.V"]
   else
     api.trace("T2V", it["T2.Sku"], it["T2.V"])
 }
```

And now these codes no longer work because aliases are returned instead (e.g. T2IP)

Regards,

Tin

How do I get the whole row object within an element of a calculation logic?

Is there a way to get the whole row object within an element of a calculation logic? I have a calculation logic which is supposed to enrich each row of a Datamart based on existing data. Currently, I use the "UserEntry" option to fetch the existing data from the row. I need to read 48 values from the row and I don't want to create additional 48 "UserEntry" elements in my calculation logic.

How can I determine which products are visible to a user?

When I create a quote, I can see only a limited set of products. Which settings in Pricefx should I check?

How to add a string to a workflow email

Is there a possibility to add text as a string to a workflow email? I tried .setReason, but it did not work out as expected.

What I am looking for is a possibily to edit the header of the email or the content.

Cheers,

Jochen

How can I read data from an existing data source?

I want to read the FiscalCalendar data source and get the FiscalData for today's date.

I tried using the api.find method but I get an error stating *Could not find property 'attribute1' in the class net.pricefx.domain.DMDataSource"def fiscalDate = api.find("DMDS",Filter.equal("uniqueName","FiscalCalendar"),Filter.equal("attribute1","20170714"))*.

How do I find a specific value in a line item?

I am looking for the following value: a specific Rebate Agreement – Line Item – Sales Targets. In the current agreement, I want search for other agreement, find the related line item (only 1 line item is added) and find the user input „Sales Targets" in this line item. I already know the Rebate Agreement number where I want search for the Sales Targets, but how to get to the Sales Target?

How do I get inputs/outputs of quote line items?

I am trying to get quote line item details using `api.find("QLI")` in a Calculation Flow logic but I am getting a simple quote object such as:

```
{version=0, typedId=47574.QLI, dirty=false, lineId=dQO02lmsmWA8ZZO,
calculationStatus=10, editabilityStatus=0, label=2M CLEAR
BOPP/FA20B/40#, sku=CLSCLS.COMMON10576, folder=false, treeLabel=2M CLEAR
BOPP/FA20B/40#, createDate=2016-12-20T19:54:15,
lastUpdateDate=2016-12-20T19:54:18, id=47574}
```

Is there any other way to get inputs/outputs of the line items in CF please?

Does the Competiton Data table have a limited number of columns?

How to use shared cache

Hi everybody,

I want to store matrices in the shared cache. It looks like I have an issue with fetching them.

I do the following:

1. Change the map to a string.

   ```
   finalMatrix = api.jsonEncode(finalMatrix)
   ```

2. Store the string with the tableName.

   ```
   def tableName = (configSister + ":" + hwg).toString()
   finalMatrices = api.setSharedCache(tableName, finalMatrix)
   ```

3. Fetch the string from the shared cache.

   ```
   if(finalMatrices != null){
    finalMatrix = api.getSharedCache(tableName)
   }
   ```

4. Change the string to a map again.

   ```
   finalMatrix = api.jsonDecode(finalMatrix)
   ```

With changing it back to a map I think there is a problem. If I want to fetch a specific line from it afterwards with `finalMatrix.findRow("Product ID", productId)` it breaks :-/

Would be great if you can help me,

Cheers,

Jochen

How to create my own currency?

In this article How to Set up Currencies you describe how to use currencies. Is there a possibility to create and deposit an "own" currency and select it from the currency drop down menu?

Why in Dashboard view the date filter (DD/MM/YYYY) only accepts day numbers < 20?

Same problem occurs also in Dashboard Configuration view, where we can set a predefined date filter to the dashboard. The calculation logic connected to the dashboard uses this groovy element:

api.dateTimeUserEntry("Date")

How come when I api.delete something that I can still api.find it?

I ran this in a test drive with allow object modifications turned on and it successfully deletes the item, but api.find still returns it.

How do I get started with PromotionManager?

Getting started with PromotionManager based on the current online material is difficult. Is there some way to learn from example cases or other material?

Cheers

How can I manipulate with datetimes?

I am working with the following date format:

def targetTimestamp = api.targetDate()?.format("yyyy-MM-dd") + 'T' + (new Date()).format("HH:mm:ss")

How can I subtract or add 5 minutes?

Cheers,

Jochen

Why is the manualResultPrice not changed until the next calculation?

I observed the following PGI behavior connected to manual overrides: when the ManualPriceExpired goes 0>1 during the calculation, the manualResultPrice is not changed until the next calculation... I've checked the source code and I wonder if those two conditions should be in a reverse order?

AbstractPriceGridCalculationTask.java - public void processPGIResult

```
if(pgi.getManualPriceExpired())
 pgi.setManualResultPrice(null);

if(expireManualResult && pgi.getManualResultPrice() != null)
 pgi.setManualPriceExpired(true);
```

If a field in the LPG is manually changed, how to trigger a line recalculation?

Hi everybody,

I have manually editable fields in an LPG (dropdowns). I want to have a line recalculation after a change. How can I trigger this?

Cheers,

thanks for your answers in advance,

Jochen

How do I tell which fixes from the change log made it into a specific build?

Pricefx provides a change log to make it transparent which bug fix / enhancement requests are included in a particular build:

- https://nexus.pricefx.net/pricefx-master-changelog.html

How can I tell which fixes and enhancements from this logs are included in a currently active version of PFX?

How do I get today's date in a Groovy script?

I found the method `api.calendar()` but it returns the date set for a quote. Only if the date is not set, it returns today.

How to add a manual editable attribute in a LPG, based on restricted fields (like the "option()" in Java for quotes?

Hello all together,

I want to add an attribute to an LPG, which allows the user to choose from a dropdown, valid for this specific element.

The manual overwrite is not the issue but the limitation of field values.

Hope somebody can give me a hand,

Cheers,

Jochen

How to "restart" aborted calculation of PL

What possibilities we have in case of the PL calculation is aborted? How to restart it?

When working with the Excel Client, what is the worksheet size limit?

How to delete Price Lists from logic?

I would like to delete Price Lists from within a scheduled logic but PLs cannot be deleted with api.delete. Is there another way I can do this?

Do you have any sample Groovy code how to update a PriceGridItem record from CFS?

Where can I call api.sendEmail?

Is there any limit on the size of an attachment to a contract?

Is there a way to make fields in PX or PP tables not editable?

How can I fill anything into the Description column inside the Calculation results section?

Whatever I have in as a description of the element, is not copied here.

Which elements must be selected in LPG as output elements?

How do I throw an exception?

Just throw a new Exception("Xxx")? That gives me Unexpected type: java.lang.Exception. Is there some exception class I can use for this?

How can I implement a relay state that defines a specific locale?

Is PFX compatible with a tablet or mobile devices?

Can I use the Groovy library in PA?

I get an error when trying to use such logic in a chart, but in the test drive it works without problems.

I can't see prices/dates in the Datamart

I have all the data in data sources and everything seems to be set up properly, but the prices in the Datamart just don't show up.

How can I avoid label resetting within sync DSs?

When I sync a customer or product data sources to the master, sometimes all of the labels get reset to be the same as the names.

When to use Product Extension and when Price Parameter?

How to decide when to make a new table as a Product Extension or as Price Parameters?

What drives such decision?

How to find Groovy functions in Confluence?

When searching Confluence for "addOrUpdate" function, you might not get the result you want.

For example, if the page only contains Groovy code excerpt "api.addOrUpdate()", then searching Confluence for "addOrUpdate" string will not find the page.

So how to run the search?

How can I export and import all data and metadata from a partition?

This is helpful for everybody who wants to:

- make a backup of their setup of the partition
- set up quickly a new partition with some pre-defined/default setup and data
- copy partition data and metadata to another partition

How to debug strange application behavior

How can I set a warning on a price list item in Groovy

How I can resolve 'too many new instances created: 503" error?

I'm getting an error that says 'too many new instances created: 503". I'm doing a lot of lookups in this logic element so I think this has something to do with it.

How can I get better search results here?

Why are some cells in my Price List grayed out?

How to Report an Issue in JIRA

The Jira application is used in Pricefx to report bugs and enhancements. When you create a new issue in Jira, we kindly ask you to provide all of the information mentioned below. A correctly created ticket in Jira is faster to handle for everyone involved in the process – product managers, engineers, QA, consultants, documentation.

To create a new issue in Jira:

1. Make sure you have an account from Pricefx and can access Jira.
2. Go to https://pricefx.atlassian.net and from the hamburger menu on the left select JIRA.
3. Click the 'Create' button in the top bar.
4. Fill in the following fields:
    a. **Project** – Enter the customer/development project on which the issue was found.
    b. **Issue Type** – Select either Bug or Improvement (the types may vary per project).
    c. **Summary** – Write a short description of the issue, e.g. "Column headers not updating with new label"
    d. **Priority** – This field combines two factors: how much the issue impacts the customer and how fast a fix should be provided.
        - **Critical** – Issues representing security threats, possible data loss, may cause system crashes etc. Should be resolved ASAP.
        - **High** – Issues with serious impact on functionality, not working features, unfulfilled requirements etc. Should be resolved in the next patch/release (depends on agreement of stakeholders).
        - **Medium** – Issues influencing functionality, but a workaround is available or the affected system part is not very commonly used. The fix time is to be discussed.
        - **Low** – Issues with little impact; a workaround can be used.
        - **Lowest** – Usually cosmetic issues, typos etc.
    e. **Affect Version/s** – Select the version(s) in which you experience the issue.
    f. **Customers** – If the issue was found in a customer environment, select the customer name.
    g. **Description** – Write detailed information about the issue. It is very useful to provide the following:
        i. Detailed description of the issue + use case.
        ii. Steps to reproduce – Step by step procedure how others can reproduce the issue. This is important for testing the issue.
        iii. Expected result / Actual result – What result did you want to achieve / What really happened. This is important for developers, so that they get the difference.
        iv. Environment – Logged user & entitlements, partition, browser + version etc.
        v. Logs, errors (if applicable)
        vi. Screenshots
        vii. Sample data (if required to reproduce the issue)

            See an example...

            Prerequisite: Submitted/approved DCR.

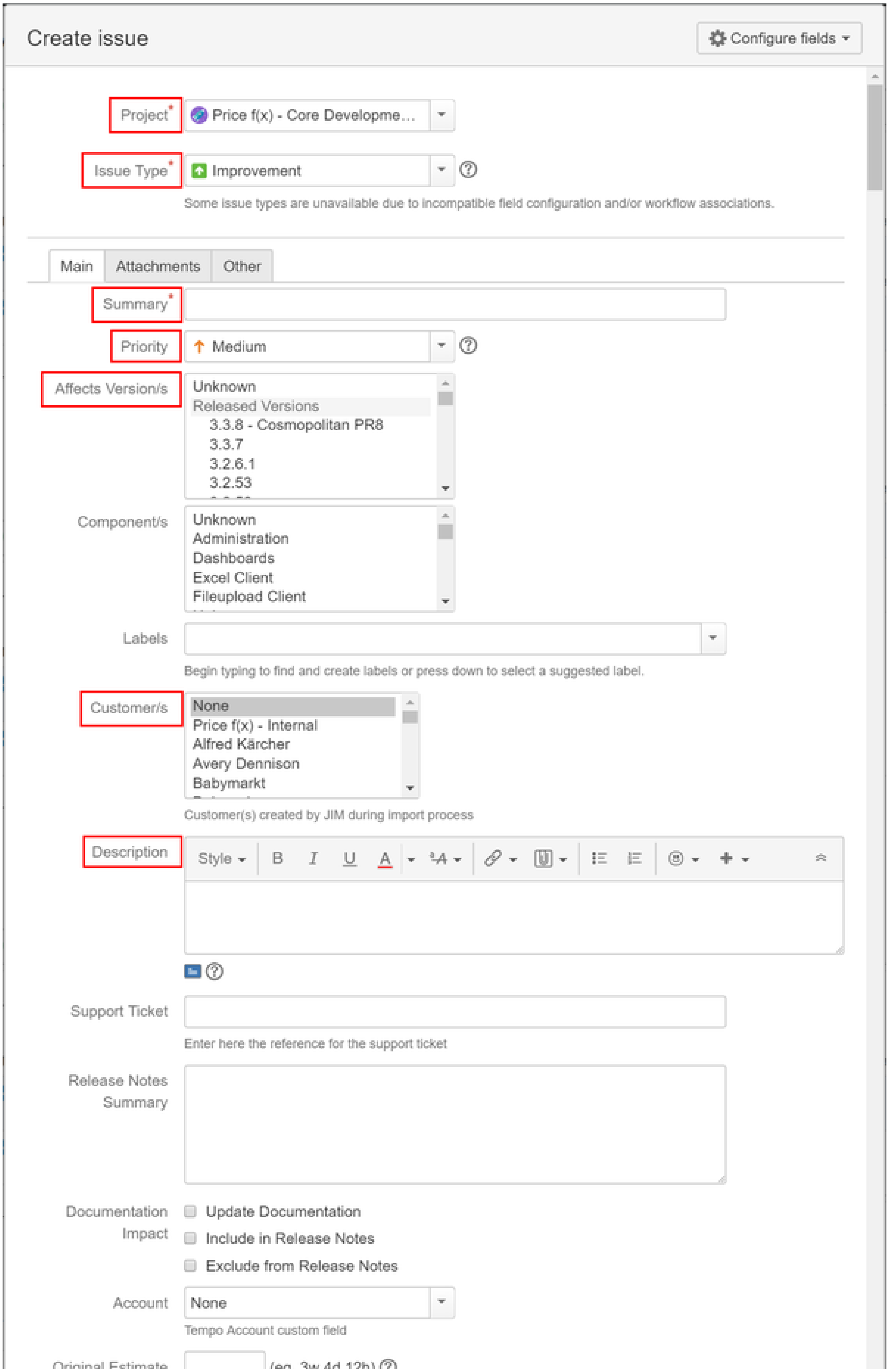
            Steps to reproduce:

            1. Go to PB > DCR and select a submitted or approved item.
            2. Try to change the value in the label field.

            Actual result: It is possible to fill in a new value even though this value isn't saved.

            Expected result: The field isn't clickable after the submitting.

5. Click 'Create'. Once the ticket is updated, you will automatically receive an email notification.



Create issue

Configure fields

Project* Price f(x) - Core Developme…

Issue Type* Improvement

Some issue types are unavailable due to incompatible field configuration and/or workflow associations.

Main | Attachments | Other

Summary*

Priority: Medium

Affects Version/s:
Unknown
Released Versions
3.3.8 - Cosmopolitan PR8
3.3.7
3.2.6.1
3.2.53

Component/s:
Unknown
Administration
Dashboards
Excel Client
Fileupload Client

Labels

Begin typing to find and create labels or press down to select a suggested label.

Customer/s:
None
Price f(x) - Internal
Alfred Kärcher
Avery Dennison
Babymarkt

Customer(s) created by JIM during import process

Description

Support Ticket

Enter here the reference for the support ticket

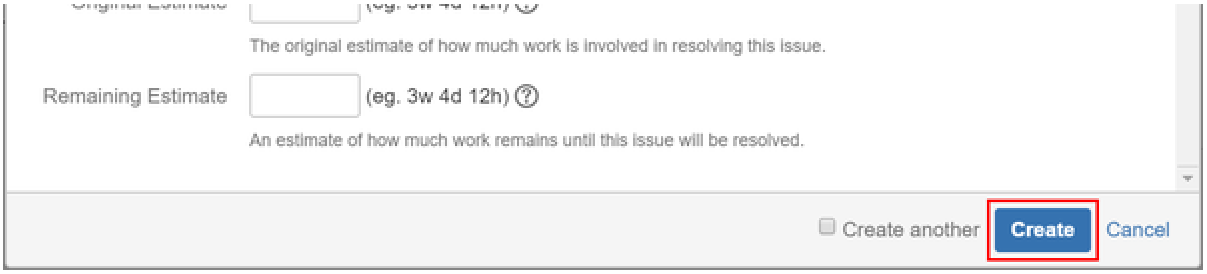
Release Notes Summary

Documentation Impact
□ Update Documentation
□ Include in Release Notes
□ Exclude from Release Notes

Account: None

Tempo Account custom field

Original Estimate (eg. 3w 4d 12h)



Original Estimate (eg. 3w 4d 12h)

The original estimate of how much work is involved in resolving this issue.

Remaining Estimate (eg. 3w 4d 12h)

An estimate of how much work remains until this issue will be resolved.

How to Format Datamart Query Results

Joining two queries below to aggregate values at an item level and at an invoice level.

What piece of code is needed to structure the output to be formatted as the example desired result:

- x = Quantity
- y = Price
- z = Order Value

Desired result:

```
[
[x:10, y:12, z:120, customer: 'Green', customer type: 'Good'],
[x:11, y:13, z:130, customer: 'Red', customer type: 'Medium']
]
```

Queries:

```
// Get datamart context
def dmContext = api.getDatamartContext()

// Create datamart query1
def dmQueryItem =
dmContext.newQuery(dmContext.getDatamart("Transactions"), false)
dmQueryItem.
select("INVOICE_NUMBER").
select("ACCOUNT_NAME").
select("LOWIEA").
select("QUANTITY_BUOM").
select("FINAL_GIVEN_NET_PRICE").
select("FINAL_GIVEN_NET_PRICE_UNIT").
where(Filter.equal("INVOICE_DATEYear","2018"),
Filter.equal("SOLD_TO", "12345678"),
Filter.equal("SALES_ORGANIZATION","12345"),
Filter.equal("MATERIAL", "XYZ"))

// Create datamart query2
def dmQueryInvoice =
dmContext.newQuery(dmContext.getDatamart("Transactions"), false)
dmQueryInvoice.
select("INVOICE_NUMBER").
select("ACCOUNT_NAME").
select("LOWIEA").
select("QUANTITY_BUOM").
select("FINAL_GIVEN_NET_PRICE").
where(Filter.equal("INVOICE_DATEYear","2018"),
Filter.equal("SOLD_TO", "12345678"),
Filter.equal("SALES_ORGANIZATION","12345"))

def joinFields = [:]
joinFields.put ("INVOICE_NUMBER", "INVOICE_NUMBER")        //Join on
Invoice#
joinFields.put ("ACCOUNT_NAME", "ACCOUNT_NAME")            //Join on
Account Name
def dmQuery1 = dmContext.newQuery(dmQueryItem, dmQueryInvoice,
joinFields, false)
dmQuery1.
select("QUANTITY_BUOM_1", "Quantity").
 //Quantity from dmQueryItem
select("FINAL_GIVEN_NET_PRICE_UNIT_1", "Price").       //Price from
dmQueryItem
select("FINAL_GIVEN_NET_PRICE_2" , "Order Value").     //Order Value
from dmQueryInvoice
select("ACCOUNT_NAME_1" , "Customer").                        //Customer
from dmQueryItem
select("LOWIEA", "Customer Type")
 //Customer Type from dmQueryItem


// Run query and save result to memory
def result = dmContext.executeQuery(dmQuery1)
if (result) {
def tableContext = api.getTableContext()
tableContext.createTable("Transaction_Current", result)
}


if (api.isSyntaxCheck()) return

def tableContext = api.getTableContext()
def query = tableContext.newQuery("Transaction_Current")
query.
select("Quantity").
select("Price").
select("Order Value").
select("Customer").
select("Customer Type")
```

```
def result1 = tableContext.executeQuery(query)
result1?.getColumnValues(0)
```

Why is SUM not working on Datamart query

The SUM function in the Datamart query below is not working as needed.

```
// Get datamart context
def dmContext = api.getDatamartContext()

// Create datamart query
def dmQueryInvoice =
dmContext.newQuery(dmContext.getDatamart("Transactions"), false)
dmQueryInvoice.
select("INVOICE_NUMBER").
select("ACCOUNT_NAME").
select("SUM(FINAL_GIVEN_NET_PRICE)").
where(Filter.equal("INVOICE_DATEYear","2018"),
Filter.equal("BILLING_TYPE","F2"))
```

Pricefx Studio

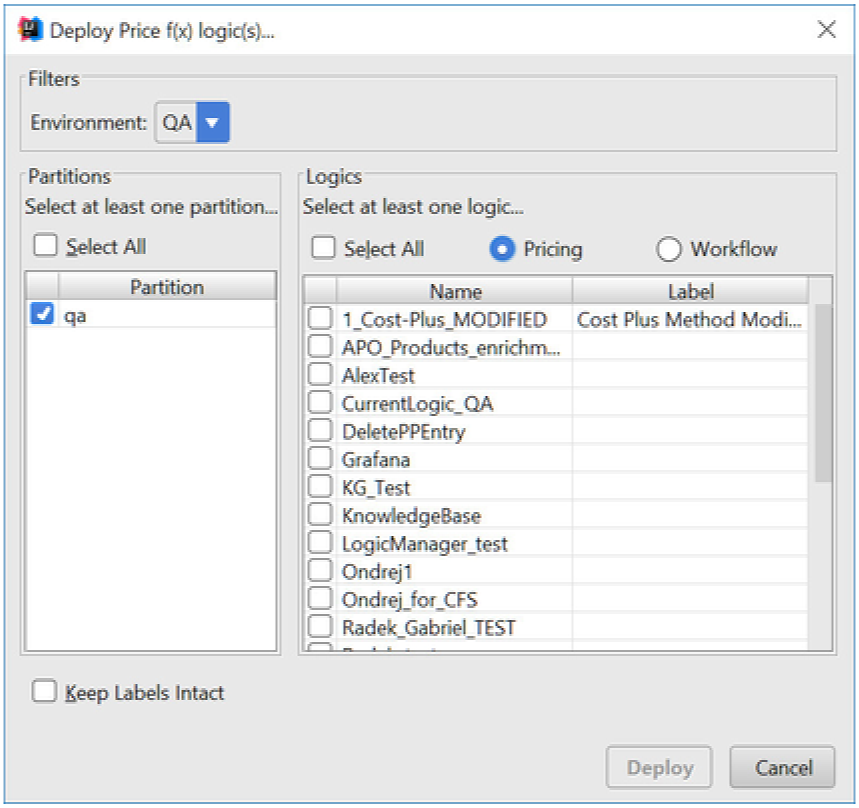
Pricefx Studio (formerly known as Logic Manager) helps you improve productivity when developing calculation logics during project implementation. With Pricefx Studio you can fetch calculation logics from remote partitions, test them and deploy them at remote partitions. Pricefx Studio runs as an IntelliJ IDEA plugin.

- Advantages
- Supported Types
- Requirements

Advantages

With Pricefx Studio, you can:

- Fetch one or more calculation logics from a selected partition.
- Deploy one or more calculation logics to selected partition(s) .
- Edit the calculation logic in IDEA.
  - For Groovy logic elements, leverage the powerful IDEA Groovy editor.
  - Make use of AutoComplete for Pricefx Public API.
  - Make use of Groovy documentation for Pricefx Public API.
- IDEA further allows:
  - Groovy variables refactoring
  - Groovy code formatting
  - Tracking changes on an element level (when storing a logic in some VCS repo)
- Test Calculation Logic execution
- Debug Calculation Logic execution (experimental)



Environment: QA

	Partition
☑	qa

	Name	Label
□	1_Cost-Plus_MODIFIED	Cost Plus Method Modi...
□	APO_Products_enrichm...	
□	AlexTest	
□	CurrentLogic_QA	
□	DeletePPEntry	
□	Grafana	
□	KG_Test	
□	KnowledgeBase	
□	LogicManager_test	
□	Ondrej1	
□	Ondrej_for_CFS	
□	Radek_Gabriel_TEST	

Supported Types

Pricefx Studio supports fetching/deployment of:

- All types of logics:
  - calculation logics (F) – must be placed in the CalculationLogic folder
  - workflow logics (WF) – must be placed in the WorkflowFormula folder
- Other PFX objects:
  - dashboards (DB)
  - price parameters (LT)
  - product attributes (P)
  - product extensions (PX)
  - customer attributes (C)
  - customer extensions (CX)
  - calculated field sets (CFS)
  - calculation flows (CF)
  - data sources (DS)
  - datamarts (DM)
  - data loads (DL)
  - preferences (PREF)
  - contract term type attributes (CT)
  - contract term types (CTT)
  - rebate type attributes (RB)
  - rebates types (RBT)
  - rebate record attributes (RR)
  - users (U)
  - user groups (UG)
  - business roles (BR)

Requirements

- Pricefx Studio runs on any Linux/Windows/Mac OS with Java 8.
- For Pricefx Studio as the IDEA plugin you need IntelliJ IDEA version 2017.3 - 2019.1.3.

Pricefx Studio Release Notes

- Latest Release
- Older Releases

Latest Release

Pricefx Studio 5.0 (build 2487)

MAJOR CHANGE

PFAUT-313 Preset parameters have been massively refactored in order to support new contexts. New virtual TypeCode parameter introduced to support this. Therefore the preset files have now a new format.

So if you need to keep the old format of your preset files, save them under a new filename. The old format can still be read.

Also after this upgrade, you should Update Project Libraries.

NEW FEATURES

- PFAUT-313 Massive refactoring of parameters to support new contexts. Test of PA logics (formulaNature "paDataLoad") is now supported. Generate or test formula now recovers parameters if a run crashes. LookupLabels fixed: they now display values as expected.
- PFAUT-344 Reusing Pricefx connections; fixed Groovy console not working behind a proxy; fixed problem with bigger HTTP requests causing 502 Bad proxy.
- PFAUT-326 Test logic now support contexts PX, C, CX, MPLI, PR
- PFAUT-226 Labels translation is now supported in logic.json editor
- PFAUT-230 Support User group field in the logic.json editor
- PFAUT-312 Added support for Advanced configurations
- PFAUT-301 Warnings of the element result are now showed in a popup dialog
- PFAUT-339 Workflow logics: dialog with approval steps can now be displayed on a workflow element result
- PFAUT-330 Updates of JavaDoc of the Groovy API from Bramble release (don't forget to Update Project Libraries)
- PFAUT-315 Make defaults false for AllowOverride and HideWarnings
- PFAUT-325 Changing texts/icons due to rebranding Price f(x) -> Pricefx
- PFAUT-342 When deploying dataloads (DL) keep Incremental Load Date when it is not present in the JSON file

BUG FIXES

- PFCD-4820 Backend now returns Map and Lists values as a String again in api.trace
- PFAUT-346 Element / trace result dialog shows now Map data type with keys unsorted
- PFAUT-309 No error message when deployment of DM fails
- PFAUT-344 Fixed Groovy console not working behind proxy
- PFAUT-343 GDSL script crash fixed. It should not break the auto-completion anymore

Older Releases

- Pricefx Studio v4 Release Notes
- Pricefx Studio v3 Release Notes
- Pricefx Studio v2 Release Notes

ℹ Pricefx Studio was released up to the version 3.2 under the name Logic Manager.

Pricefx Studio v4 Release Notes

This document summarizes major improvements and changes introduced in Pricefx Studio.

Pricefx Studio 4.8 (build 2335)

After upgrade, follow these steps: Update Project Libraries

NEW FEATURES

BUG FIXES

- PFAUT-296 bugfix: Workflow logic test execute was crashing every time
- PFAUT-304 bugfix: Auto-completion for 'lib' and 'libs' did not work if the logic contained a non-groovy element
- PFAUT-102 bugfix of searching SKU in PfxTextFieldWithAutoCompletion
- PFAUT-306 added support for Rebate agreement templates and Rebate agreement template attributes

IMPROVEMENTS

- PFAUT-299 PA SQL query console can now be used for displaying other types than Grid results - so it can be used in a similar way as the Groovy console for all functions that do not use parameters
- PFAUT-300 Added auto completion of typeCode members for api.find and api.stream
- PFCD-5053 Massive Groovy API JavaDoc enhancements. The documentation can be displayed by pressing CTRL +J (Win) or F1 (Mac). What has been updated:
  - extended with more information
  - included best practices from the configuration point of view
  - added code samples added for majority of functions (Example:)
  - deprecated a few functions (@ deprecated)
  - added links to similar functions (@ see)
  - added links to reference documentations (@ see)
  - added reference to system (=server) variables
  - added references to UI (targetDate, addWarning, …)
  - fixed various typos
- PFAUT-286 Product group type is now supported in Parameters
- PFCD-4870 Saved charts are now supported for fetch/deploy

UI CHANGES

- PFAUT-289 UI component for api.option() now shows the labels (in the Parameters tab of the logic editor)
- PFAUT-288 Save button in the element editor has been moved to the top of the panel
- PFAUT-297 Scrollbars added in the element result popup dialog

Pricefx Studio 4.7 (build 2272)

NEW FEATURES

- PFAUT-211 Added support for Price Records metadata (fetch/deploy)
- PFAUT-172 Added support for MessageTemplates (fetch/deploy)

IMPROVEMENTS

- PFAUT-102 Auto-completion for parameters (e.g. SKU) done by IDEA component
- PFAUT-256 Auto-completion for DIMFILTERENTRY parameters done IDEA component

BUG FIXES

- PFAUT-283 Fetch all stopped working in the last release
- PFAUT-186 Not possible to enter custom condition in the element condition field
- PFAUT-252 Various small fixes of header logics

Pricefx Studio 4.6.2 (build 2231)

BUG FIXES

- Generation of parameters crashed (a bug was introduced in the previous release)
- PFAUT-267 PA SQL query console does not work when a proxy server is used
- Bugfix: Groovy console evalution of api.productExtension, api.productCompetition, api.productXRef when filters were used

Pricefx Studio 4.6.1 (build 2218)

BUG FIXES

- PFAUT-275 Fixed a bug that when a logic deployment failed, the summary popup box displayed OK
- PFAUT-268 Logic testing of api.decimalUserEntry does not rreturn string anymore
- PFAUT-278 Generate parameters crashes when DS has no records

IMPROVEMENTS

- PFAUT-267 PA SQL query console improved so it supports DS/DM fields selection and generation of the result Groovy element
- PFAUT-275 Error handling reworked and unified, PFX error messages (such as "Login failure" etc.) are now propagated to the user
- PFAUT-277 Rank attribute from DS/DM field to main source.json / datamart.json file

- PFAUT-102 Load possible parameters values

MINOR ADJUSTMENTS

- The last element result is pre-selected (including the api.traces()) when running a new test.

Pricefx Studio 4.6 (build 173.2203)

BUG FIXES

- PFCD-4249 Element results and api.trace() of type Map or List are now supported by the result popup tree browser (requires the Salty Dog release)
- PFCD-3333 Error line number is now correct when calling a function from another element
- PFCD-4193 Failing element was listed twice in a test formula list when calling a function from another element
- PFAUT-245 Cannot run Workflow formula test for price lists
- PFAUT-250 Parameters generation failed when using configurators
- PFAUT-238 Deployment of Customer attributes was fixed
- PFAUT-239 Deploy of RebateTypeAttributes and RebetaRecordAttributes was fixed
- PFAUT-265 Saving results of a logic test caused corruption of logic.json
- PFAUT-255 Fixed fetching of dashboards not to be environment specific

NEW FEATURES

- PFAUT-267 PA SQL query console (available in Tools PA SQL console)
- PFAUT-112 Logics quick search (use CTRL+SHIFT+L in Windows, +SHIFT+L on Mac). On Windows, you may need to remove the secondary shortcut for "Find Previous" in Settings Keymap. On Mac, you may need to disable the global setting in System Preferences Keyboard Shortcuts Services Search with Google.
- PFAUT-246 Added auto-completion for getDataSource() and getDataMart()
- PFAUT-217 api.getElement() now alerts if the element doesn't exist (e.g. after renaming). Similarly, some other functions check the existence of PX, LT etc.
- PFAUT-231 Possibility to fetch complete configuration of a partition (= fetch all objects)
- PFAUT-229 Added option "With preferences" when fetching/deploying P, PP or PX/CX
- PFAUT-261 Debug support / Groovy console support for api.getPricelistSummaryQuery, api.getPriceGridSummaryQuery(), api.getRebateRecordSummaryQuery(), api.getSimulationSummaryQuery()
- PFAUT-262 Added auto-completion support for Calendar, Table, Column in DatamartContext
- PFAUT-256 Added options for parameter type DMDIMFILTER
- PFAUT-252 Test formula for the header logic
- PFAUT-251 Support for matrix price lists and price lists in Test formula

IMPROVEMENTS

- PFAUT-270 Groovy console output moved from the logic.json editor to the PfxResult tool window
- PFAUT-241 Refresh icon on the Fetch/Deploy panels
- PFAUT-240 Display FormulaNature in logic.json editor

MINOR ADJUSTMENTS

- PFAUT-243, PFAUT-244 Change the name of Sandbox to something more intuitive

Pricefx Studio 4.5 (build 173.2094)

This version was released mainly to fix a serious bug – generation of parameters crashed with the new Cuba Libre PR4 backend.

BUG FIXES

- Critical: PFAUT-235 Cannot generate parameters after latest PFX release
- PFAUT-181 Proxy server is not working since IDEA 2017.3.2
- PFAUT-206 Pricing parameters with references to other PP doesn't gets lost
- Result view dialog crashed if the result was not valid JSON.

NEW FEATURE

- Logic.json editor: The User group field can be viewed/edited (as a text field for now). Values are separated by "," without space.

Pricefx Studio 4.4 (build 173.2031)

This version was released mainly to fix a serious bug – crash when creating a new project.

BUG FIXES

- Critical: creating a new IDEA project with the PFX support crashed.
- The alert fields are not properly saved when editing, fetching or during deployment.
- Auto-completion of 'libs' inside the Groovy library was not working.
- Problem with configurators raising an exception was fixed.

FEATURE

- Flag allow override is now supported in the logic.json editor.
- Evaluate expression now shows results of api.findLookupTableValues() in a grid as well.

Pricefx Studio 4.3 (build 173.1979)

UPGRADE INSTRUCTIONS

Update the project libraries as described in Step 3 in Upgrading Logic Manager to Pricefx Studio.

BUG FIXES

- Metadata tool window exception thrown after opening a new project without config.groovy.
- Auto-completion did not work if the metadata had not been loaded by the Metadadata window before.
- Element copy caused formulaExpression to be saved in the logic.json. Kindly remove any formulaExpression that you see in the logic.json. It should not have a functional impact but this property should not be stored in GIT.
- Project tree did not sort elements if there was a RESULT element (=) since for that the element file was not created.
- Various issues in the Metadata tool window, e.g. reloading.
- Fields Threshold in the Alerts section of the logic.json editor did not get saved.
- Exception in the element unit test file creation if the CalculationLogicDir folder did not exist.
- Adjusted template for the element unit test file, so it does not raise errors (removed CalculationLogicDir from the LOGIC_DIR variable).

FEATURE: SANDBOX GROOVY CONSOLE

- Used to test a small part of the Groovy element (without the need to run the whole logic with all the elements).
- You can test your data returning API functions such as `api.find()`, `api.findValues()`, `api.stream()`, including filters and see the data returned by those functions in the table grid. Note that for `api.find()` and `api.stream()` a limit is applied.
- You can copy the data structure from the grid as a Groovy List<Map> source code to be able to mock your data in your unit tests. Right-click in the grid and choose "Copy Groovy source code to clipboard" from the menu.
- The list of supported functions are listed in Debugging a Logic in Pricefx Studio in IDEA, since it internally uses the debugger functionality. Function `api.getElement()` is not supported.
- You can test your Groovy library method calls before deploying them on server (if the method uses the above supported methods).

FEATURE: KEYBOARD SHORTCUTS ADDED - YES, FINALLY! :)

Alt + F5	Runs the logic test. Works from the logic.json editor as well as from the element source code editor.
Alt + 8 Windows Cmd + 8 Mac	Fetch window
Alt + 0 Windows Cmd + 0 Mac	Deploy window
Alt + F8	Evaluates expression from the Sandbox.
Ctrl + A	Selects all logics in the Fetch and Deploy tool windows.
Tab	Supported in the logic.json editor.

These shortcut keys can be personalized – navigate to menu Settings (Windows) or Preferences (Mac) > Keymap. Search for: "Run logic test", "PfxFetch", "PfxDeploy".

OTHER FEATURES

- Translation of formulaNatures (diferrent types of logics) in the new logic creation dialog to a human readable language.
- Environment/partition selector is now added in the Metadadata tool window. It also tells the auto-completion which partition metadata to use.
- The folders CalculationLogic, WorkflowFormula, CalculationLogicTest are marked as Sources root / Test sources root when performing "Add framework support".
- Added element numbers in the logic.json editor.
- Added lookup label for the Customer parameter in the parameters editor.
- More detailed error message when deploying product extensions.
- Added support of more functions in the debugger (since Cuba libre).

- Updated JavaDoc based on Cuba Libre.

Pricefx Studio 4.2 (build 173.1936)

UPGRADE INSTRUCTIONS:

Update the project libraries as described in Step 3 in Upgrading Logic Manager to Pricefx Studio.

BUG FIXES

- PFAUT-189: Fixed problem with loosing the labelTranslations element property when editing the elements properties.
- PFAUT-194: Fixed problem with saving of non-groovy elements of types ADD, SUBTRACT, RESULT or BRICK which were save to files with the suffix ".formula" but instead the correct suffixes should have been: ".add", ".subtract", ".result" and ".brick".
- The element duration information stopped working due to a changed format of the __TRACE__ element (after a new column was added in the Cosmopolitan version).
- Auto-completion of PFX metadata (attributes) was not working if the Metatada tool window was not opened before (and populated the meta database).
- PFAUT-198: The lookup labels for logic parameter values in the Parameters tab in logic.json didn't take the environment/partition change into consideration.

NEW FEATURES AND CHANGES

- The automation library and Swagger HTTP client have been completely removed from the engine and replaced everywhere with the pfxpackage tool (and its own faster HTTP client). As a result of that, the size of the installation package has decreased by almost 50%, so the upgrades are faster now and the memory/time needed for file indexation done by IDEA is reduced.
- The internal model how Studio works with the logics and elements data has been changed in order to be able to work with the future Pricefx releases without the need of upgrading Studio (e.g. when a new property is added in a new Pricefx release).
- It is now possible to create all types of logics in Studio (= with the corresponding formulaNature). Formerly only pricing and workflow logics could be created.
- PFAUT-197: Editing of the Calculation Context element property is now possible from the logic.json editor (the elementGroups field in JSON).
- The class com.googlecode.genericdao.search.Filter has been moved to the default package to avoid automatic creation of 'import' statements.

Pricefx Studio 4.1 (build 173.1889)

If you would like to use this version and still have a version lower than 4.0, follow the steps in Upgrading Logic Manager to Pricefx Studio.

NEW FEATURES AND CHANGES

- The legacy Logic Manager core has been redesigned (the engine based on an automation library has been replaced with the pfxpackage engine).
- Added support for fetching/deployment of other PFX objects: LT, P, PX, CX, CFS, CF, DS, DM, DL, DB, PREF, CT, CTT, RB, RBT, RR, U, UG, BR.
  Prerequisite: all non-workflow logics must be located in the CalculationLogic folders and workflow logics must be placed in the WorkflowFormula folder.
- It is possible to deploy multiple objects to multiple partitions.
- Error reporting has changed: deployment errors now appear in a popup window. The error log has been moved from the PFX Console to the Event log.
- Removed dependency on a particular Groovy library version: you can use the same plugin version again for both versions of IntelliJ IDEA 2017.2 and 2017.3. So using versions 173.xxxx is 2017.2 compliant.
- In order to fetch logics with different validAfter, they are fetched to a folder having the validAfter value as a suffix e.g. `Current_Logic+2017-02-23`. The suffix can be removed afterwards, it is not used for anything. The logic's name and validAfter are determined now only from logic.json.
- Bugfix: the syntax of the newly created config.groovy has been fixed (2FA 'null' => null).
- Auto-completion support was improved. Added support for `libs` (multiple Groovy libraries). PFX binded variables (actionBuilder, quote, raProcessor etc.) are now suggested only in logics where they are applicable.

Pricefx Studio 4 (builds 172.1814, 173.1815)

To upgrade to this version, follow the instructions in Upgrading Logic Manager to Pricefx Studio.

> If you have configured proxy servers in IDEA, you can upgrade your IDE up to the version 2017.3.1, but not higher for the moment!

> Since this version, the plugin repository URL has been changed. You can find it at Pricefx Studio Download URL.

MAJOR CHANGE

- IDEA 2017.3 is now supported.
- Plugin is now available in two versions: for IDEA 2017.2 and 2017.3 (due to different Groovy versions).
- Plugin version numbering has been changed due to the point above.
- Plugin name has changed from Logic Manager to Pricefx Studio.
- Logic format as used in the Pricefx Cosmopolitan version is now supported (labelTranslations).
- API sandbox Javadoc and auto-completion updated to the Cosmopolitan version.
- PFAUT-159 – Added support for Inline configurators.
- PFAUT-158 – Added support for Condition and Suffix in the formula element editor.

NEW FEATURES

- PFAUT-165 – It is possible to generate a TDD Test template by clicking in a second column in logic.json
- PFAUT-162 – The 'Allow object modification' option from the formula test has been implemented.

UI CHANGE

- Buttons in various panels have been moved to the left so they don't get hidden when resizing.

BUGFIXES

- Crash exception has been fixed when the plugin was installed in a directory path that contained a space.

Pricefx Studio v3 Release Notes

This document summarizes major improvements and changes introduced in Pricefx Studio.

- Pricefx Studio 3.2 (build 173.1783)
- Pricefx Studio 3.1 (build 1631)
- Pricefx Studio 3.0 (build 1492)

Pricefx Studio 3.2 (build 173.1783)

MAJOR CHANGE

- IDEA version 2017.3 or higher is required!

Pricefx Studio 3.1 (build 1631)

MAJOR CHANGE

- IDEA version 2017.2 or higher is required.

NEW FEATURES

- The element duration time is shown in results for each element. Moving mouse over will display a tooltip with detailed information from the "__TRACE__" element. This helps you find potential performance issues already during the development phase.
- Two Factor Authentication is now supported. The token needs to be saved in config.groovy by the 'Create new Pricefx logic' dialog.
- The 'Create new formula' dialog now supports workflow formulas too.
- It is now possible to open an element source code by double clicking or by pressing the SPACE key in the element list.
- Elements can be copied now.
- The element property 'Hide warnings' is now shown and can be edited.
- Support of configurators in the formula test has been improved.
- The target date parameter has been added to the formula test.

BUG FIXES

- Element editing always failed (since the last release).
- Meta data toolbox didn't get loaded in some cases. Improved error handling (e.g. "No access to command" or no network connection). Errors when loading data from PFX do not cause a complete stop, but display only information that is available.
- Couldn't edit element's properties if there's only one logic element.
- Logic element was deleted even Cancel was chosen.
- Valid result of type Map was not parsed and displayed in the Result viewer tree.

OTHER IMPROVEMENTS

- Added various fields validation (element name, date fields).
- The Save button moved to the left not to be hidden by the Tool windows.

Pricefx Studio 3.0 (build 1492)

MAJOR CHANGE

The format of logic.json has been slightly changed. It does not include the JSON properties with the value 'null' anymore. This change is backward compatible, so you do not have to do anything. Only after fetching a logic which was stored in GIT you will get a diff which you should commit in order not to get it again.

The attribute ResultGroups for an element is now supported in fetch/deploy and in the editor. If you are using ResultGroups in your logics, you should upgrade to this version straight away.

NEW FEATURES

- Deploying logic does not perform a dry run anymore, so the deployment should be faster and you do not need api.isSyntacCheck() anymore.
- Metadata inspector will help you list or search the attributes of PFX objects that you use in api.find.
- To be prepared for usage of the new pfxpackage tool in the future, the default fetch logic directory has been changed to <project_root>/CalculationLogic for the newly created projects. The existing ones can stay as they are, however it is recommended to migrate them manually – by moving all logics to the CalculationLogic subfolder.
- Added auto-completion for the 'cProcessor' variable (Contract builder).
- You can write automated logic tests that will perform a test run and compare the elements' result values with the ones saved manually after running a test run manually (use the button Save results). You can define the elements whose result values should be used in the comparison. The template of a unit test can generated by typing `pfxtest<enter>`. You will need to add Maven support in your project and add the Maven dependency in the Spock framework in pom.xml. The documentation of the Spock framework is here.

OTHER IMPROVEMENTS

- The element edit button was removed. Element properties are now displayed in the write mode by default.
- It is possible to add more environments and partitions using the Create Pfx(x) configuration dialog.
- The logic name can be refactored in IDEA. Rename the logic using 'Refactor rename' (SHIFT+F6) and that will adjust uniqueName in logic.json. "Search for references" must be ticked in the dialog though.
- Since the Mojito 3.2.50 release, the test run error will navigate you to the occurrences also for the Groovy library elements. Just click the red icon next to the element name.
- Bug fixes:
  - Pricefx Studio crashed when the config file did not exist.
  - Font style in the result matrix grid was not applied.

Pricefx Studio v2 Release Notes

This document summarizes major improvements and changes introduced in Pricefx Studio.

Pricefx Studio 2.3 (build 1240)

MAJOR CHANGES

- Only IDEA 2017 and higher is now supported.
- The way Pricefx Studio works with files was significantly reworked: the Pricefx Studio plugin now uses the internal virtual file system in IDEA, so there is no need to push the 'Sync' action to get files.
- You can get the most recent JavaDoc if you apply 'Add framework support' > Pricefx to the existing project.

ADDED AUTO-COMPLETION SUPPORT FOR:

- `lib` – Injected variable for Groovy library elements. The __LIBRARY__ logic must be present on the disk and its folder must be part of "Sources" in the module setting.
- `api.product()` – Lists the possible Product fields + their labels defined in PFX.
- `api.inUserGroup()` – Lists the user groups defined in PFX.
- `api.findDataLoad` – Lists the data loads defined in PFX.
- `api.getManualOverride(), api.removeManualOverride()` – Lists available elements.
- `api.vLookup, api.findLookupTable(), api.findLookupTableValues()` – Lists the lookup tables defined in PFX.
- `api.pricelist(), api.pricelistItem(), api.getPricelistItem()` – Lists the price lists defined in PFX.
- `api.customerExtension(), api.productExtension()` – Lists the extensions defined in PFX.
  For the auto-completions with connection to Pricefx defined above, the first connection in config.groovy is used. The values are fetched only the first time and cached.

- `api.find(), api.count(), api.add(), api.addOrUpdate(), api.update(), api.delete(), api.stream(), api.emitPersistentObject()` – Lists available type codes.

- `actionbuilder` injected variable
- `quoteProcessor` injected variable
- `raProcessor` injected variable
- `rebateRecords` injected variable
- `quoteProcessor` injected variable
- `ConfiguratorEntry` object
- `ContextParameter` object
- `DashboardController` object
- `DatamartContext` object
- `DatamartRowSet` object
- `ResultFlexChar` object
- `ResultGauge` object
- `TableContext` object

OTHER NEW FEATURES AND CHANGES

- Pricefx API JavaDoc was updated to the last version. Requires re-applying 'Add framework support' > Pricefx to the existing project.
- Elements can be sorted now.
- Recently used environment and partition is now remembered in the Fetch/Deploy tool windows whenever the config.groovy file is changed.
- Element timeout is now added in the edit element form.
- Editing of an element does not do unexpected changes of: null => empty string in logic.json anymore.
- Logic name is now shown as the title in the editor of the logic.json file.
- Debugging as a beta version added. Requires re-applying 'Add framework support' > Pricefx to the existing project.

LIVE TEMPLATE SUPPORT

Live template support was added.

- Typing "pfxfind" <ENTER> will paste a common code snippet for api.find().
- Typing "pfxquery" <ENTER> will paste a common code snippet for writing a DM query using api.getDatamartContext().

More functionality will be added later. You can create your own templates in File > Editor > Live Templates > PriceFx.

Pricefx Studio 2.2 (build 962)

> There is a major change of behaviour in the deploy action: it now takes the 'Valid after' date into consideration, whereas previously the latest active logic from the target environment was used.

EXTENDED SCOPE

Pricefx Studio newly supports:

- Formula test of Workflow logics, rebates and contract.
- Fetching logics with different 'Valid after' dates or with 'Status' set to Inactive.
  ⚠ Keep in mind that logics with different 'Valid after' dates are fetched to the same logic directory, so make sure you fetch one valid logic at a time!

IMPROVEMENTS IN USER INTERFACE

- Added buttons for filtering logics based on their formulaNature (pricing logics, Groovy library, matrix, CFs, quotes, PA, rebates, contract, workflow).
- Logic's 'Valid after' is now shown in the logic's list in the fetch tool window.
- Logic's 'Name', 'Valid after' and 'Status' are now shown in the logic.json editor.

BUGFIXES

- Object browser crashes when viewing results of a Map class.

Pricefx Studio 2.1

IMPROVEMENTS

- The actions Fetch and Deploy can now be permanently displayed on the right side of the IDEA window, so that you can access them faster whenever needed. By default, they are hidden; to have them displayed, go to the View menu > Tools Buttons.
- There is better support for entering the test parameters. On the Test Logic tab there is a new Parameters sub-tab where you can generate parameters, save them as a preset or have them pre-filled with values from the last run. As a consequence, the params.json file is no longer used/created.

Pricefx Studio 2.0

CHANGES

New Structure of logic.json

The logic.json file has a slightly different structure – but this change is handled by Pricefx Studio. When you open logic.json with the old format, Pricefx Studio converts it to the new format automatically. The only impact is if you use a versioning system (GIT, CSV, ...) – then you have to commit logic.json with the new structure.

The change is that the following part was removed:

```
{
"response" : {
"data" : [
<-- only the content between these arrows remains from now on -->
],
"status" : null
}
}
```

IMPROVEMENTS

Improvements in Logic Testing

Pricefx Studio newly supports:

- Workflow formulas as a logic type to be fetched and tested.
- Matrixes and hyperlinks as result element types.
- Alerts (critical/red/yellow) which can be defined for an element.

Improved usability:

- There are new icons to add, edit or remove elements; you don't have to do these changes in the code manually.
- It is possible to use a quick search field for a list of elements.
- A red icon is shown for an element that had a parse or runtime error. Clicking on this icon takes you to the exact place where the error was found. (This functionality requires the Mai Tai version.)
- A new object browser that displays structure of the element results or element traces that are arrays and objects (like the IDEA debugger panel).
- If the targetDate parameter is not defined in params.json, Pricefx Studio will automatically set it to the current date.

Improved navigation:

- Elements can be opened from test results right away.
- The virtual element __TRACE__ is now displayed as well.
- There are icons for each element type displayed in logic.json.

Improved look & feel:

- Space on the screen is better utilized.
- Column sizes in element result traces better correspond to their content.
- There is a better look & feel thanks to using the native IDEA UI components.

Performance Improvements

- Logic deployment is now faster.
- Logic test run is faster thanks to having the logic parsing optimized.

Update Project Libraries

This activity is recommended to be performed for every existing project on your local drive after every upgrade of the plugin. It applies the latest documentation and auto-completion in your existing projects:

1. Open the project in IntelliJ IDEA.
2. In the project files tree, right-click the root folder and choose 'Open Module Setting'.

3. Click 'Libraries', select all in the middle and remove 'Pricefx API' and 'Pricefx debug' using the 'minus' button.
4. Click OK to close the Project Structure dialog.
5. In the project files tree, right-click the root folder and choose 'Add Framework support...'.
6. Select 'Pricefx' and click 'OK' to close the dialog.

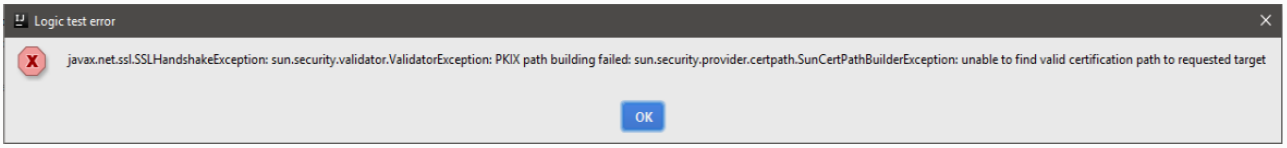
Installing Pricefx Studio IDEA Plugin

Pricefx Studio is installed through Enterprise Plugin Repository by following these steps:

1. In IntelliJ IDEA Windows version, click 'File' and select 'Settings'. In Mac version, click 'IntelliJ IDEA' and select 'Preferences'.
2. Open the Plugins tab.
3. Click the Browse Repositories button.
4. Click Manage Repositories button
5. Click the 'plus' button to add a new custom repository.
   To get the repository URL, go to this page (registered users only).
   Optionally, click 'Check now' to see that the URL can be accessed.
6. In the list of plugins, search for *pricefx* to find the plugin named *pricefx-studio-idea.*
7. Select it and click 'Install'.
8. Restart IDEA (as asked by the system).

When a new version of the plugin is released, IDEA will check during startup and offer you to update the plugin version.

Troubleshooting
If you connect to a custom Pricefx server, you might get an error message "unable to find valid certification path to requested target":



javax.net.ssl.SSLHandshakeException: sun.security.validator.ValidatorException: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target

In that case you need to import the server certificate to the certification store of JDK/JRE.

Export the SSL certificate of the Pricefx server that you are accessing in your browser, e.g. via Firefox:

1. Click the lock icon next to "https://...." and click "More details" to display the page security dialog.
2. On the last tab "Security" choose "View certificate".
3. On tab Details choose "Export". Select the "X.509 (DER)" format and save the file pfx.der into your home directory.

Then in JDK of IntelliJ IDEA run this (search the cacerts file and adjust the path jre/lib/security/cacerts accordingly):

Mac:

```
cd "/Applications/IntelliJ IDEA CE.app/Contents/jdk/Contents/Home/"
sudo keytool -import -alias pfx -keystore jre/lib/security/cacerts -file ~/pfx.der
```

Win (use importcert instead of import. Also adjust your installation path of IDEA):

```
cd "c:\Program Files (x86)\JetBrains\IntelliJ IDEA Community Edition 2017.1.4"
bin\keytool -importcert -alias pfx -keystore jre\lib\security\cacerts -file pfx.der
```

Then enter "changeit" (without the double quotes) as the password to the JRE cert store.

Pricefx Studio Download URL

This page is part of a space which completely public but for security reasons this page is accessible only for our registered users (PFX, customers, partners). Anonymous users have no access here.

When approached by customers/partners without a Confluence account, it is OK to share the download URL with them.

When installing Pricefx Studio, use this repository URL: https://developer.pricefx.eu/studio/

Previous Releases of Studio

Previous releases should generally not be used. However, if there is an issue with the upgrade to the latest version, you can manually install a previous version.

1. Download the previous Studio version to your download directory, **do not unzip**.
2. In the IntelliJ IDEA Windows version, click 'File' and select 'Settings'.
   In the Mac version, click 'IntelliJ IDEA' and select 'Preferences'.
3. Open the Plugins tab.
4. Search for "pricefx-studio-idea" and click "Uninstall".
5. Click on "Install plugin from the disk..." and choose the ZIP from the first step.

IntelliJ IDEA 2017.3

- 173.2335
- 173.2272
- 173.2231
- 173.2203
- 173.2094
- 173.2031
- 173.1979
- 173.1936
- 173.1889
- 173.1815

IntelliJ IDEA 2017.2

- 173.1889
- 172.1814

Upgrading Logic Manager to Pricefx Studio

Step 1: Upgrade Projects

First it is necessary to make a change in every project started in IDEA in the previous version of the plugin (i.e. Logic Manager) – you need to move the logics into a new subfolder:

1. Open the project in IntelliJ IDEA.
2. In the project files tree, right-click on the root folder and choose New Directory.
3. Enter *CalculationLogic* (exactly like this) as a name and click 'OK'.
4. Select all your non-workflow logics and drag them to this new *CalculationLogic* subfolder.
5. In the project files tree, right-click on the root folder and choose New Directory.
6. Enter *WorkflowFormula* (exactly like this) as a name and click 'OK'.
7. Select all your workflow logics and drag them to this new *WorkflowFormula* subfolder.
8. Commit the changes in GIT.

> Moving the logics to the *CalculationLogic* and *WorkflowFormula* subfolders does not have any impact on your team members who are still working in the legacy Logic Manager, since the legacy Logic Manager supports that. So you can do that any time.

Step 2: Upgrade IDEA plugin

1. In the IntelliJ IDEA Windows version, click 'File' and select 'Settings'. In the Mac version, click 'IntelliJ IDEA' and select 'Preferences'.
2. Open the Plugins tab.
3. Search for *pricefx* to find the plugin named *pricefx-logic-manager-idea* and click 'Uninstall'.
4. Click the Browse Repositories button.
5. Click the Manage Repositories button.
6. Select the repository URL and click the 'minus' button.
7. Click the 'plus' button to add a new custom repository.
   To get the new repository URL, go to this page (registered users only).
   Optionally, click 'Check now' to see if the URL can be accessed.
8. In the list of plugins, search for *pricefx* to find the plugin named *pricefx-studio-idea.*
9. Select it and click 'Install'.
10. Restart IDEA (as asked by the system).

Step 3: Update Project Libraries

Apply the latest documentation and auto-completion in your existing projects:

1. Open the project in IntelliJ IDEA.
2. In the project files tree, right-click the root folder and choose 'Open Module Setting'.
3. Click 'Libraries', select all in the middle and remove them using the 'minus' button.
4. Click OK to close the Project Structure dialog.
5. In the project files tree, right-click the root folder and choose 'Add Framework support...'.
6. Select 'Pricefx' and click 'OK' to close the dialog.

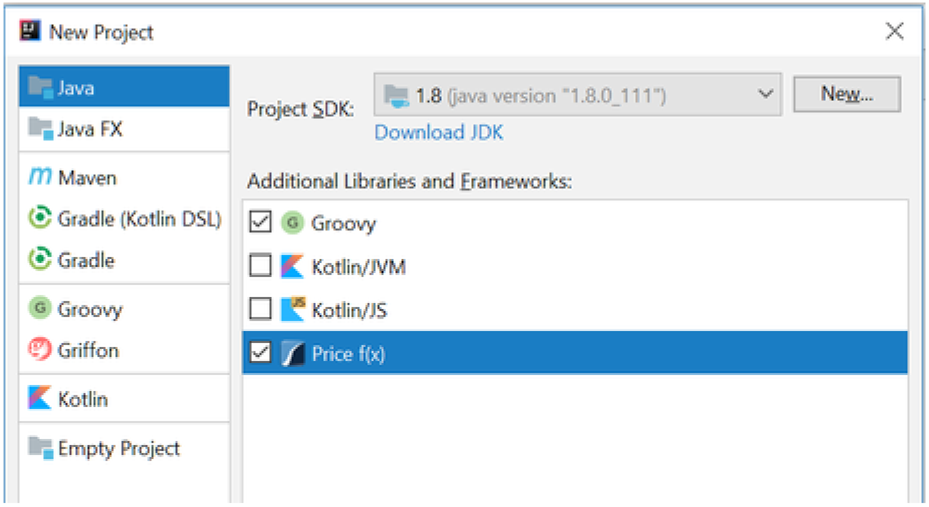
> This change should not have any impact on your team members either, since the *.idea* subfolder is usually set to be ignored by GIT.

> This step is recommended to be performed after every upgrade of the plugin.

Creating a New Project in Pricefx Studio in IDEA

To create a new project with Pricefx support:

1. Click File > New > Project.
2. Select 'Java' as the project type.
   The item Pricefx should be displayed under 'Additional Libraries and Frameworks'.
3. Select items 'Groovy' and 'Pricefx' as additional libraries for the newly created project:

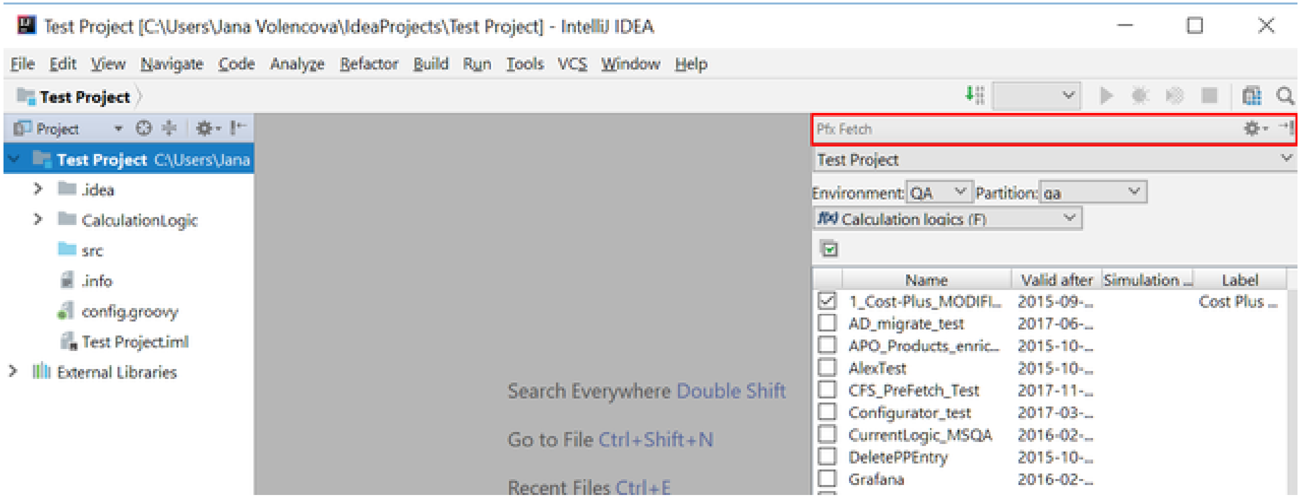


Project SDK: 1.8 (java version "1.8.0_111")

Additional Libraries and Frameworks:

☑ Groovy
□ Kotlin/JVM
□ Kotlin/JS
☑ Price f(x)

4. Click Finish to create the new project in the selected folder.
5. Make sure you have the Pricefx actions displayed.
   In the View menu check the option 'Tools Buttons'. This will display the two main actions – Fetch and Deploy – on the right side:


Test Project [C:\Users\Jana Volencova\IdeaProjects\Test Project] - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Test Project
Project
Test Project C:\Users\Jana
.idea
CalculationLogic
src
.info
config.groovy
Test Project.iml
External Libraries
Search Everywhere Double Shift
Go to File Ctrl+Shift+N
Recent Files Ctrl+E
Pfx Fetch

Test Project

Environment: QA Partition: qa

Calculation logics (F)

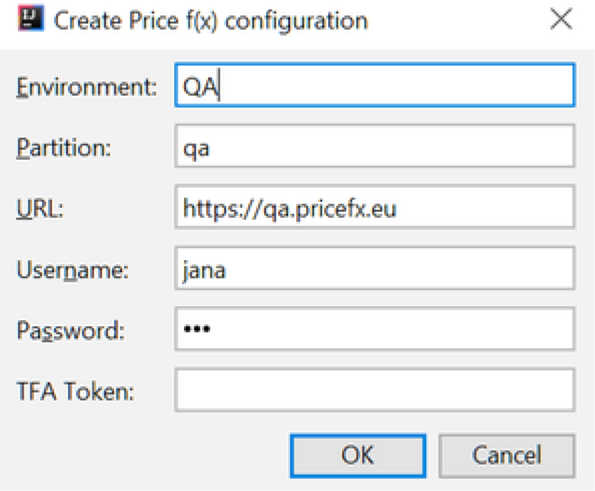
	Name	Valid after	Simulation ...	Label
☑	1_Cost-Plus_MODIFI...	2015-09-...		Cost Plus ...
□	AD_migrate_test	2017-06-...		
□	APO_Products_enric...	2015-10-...		
□	AlexTest	2015-10-...		
□	CFS_PreFetch_Test	2017-11-...		
□	Configurator_test	2017-03-...		
□	CurrentLogic_MSQA	2016-02-...		
□	DeletePPEntry	2015-10-...		
□	Grafana	2016-02-...		

Creating a Config File in Pricefx Studio in IDEA

To be able to use Pricefx Studio in IDEA, you need to have a configuration file.

To create a config file:

1. Right click the project name and from the context menu select 'Create Pricefx configuration'.
2. Fill in the environment, partition, URL, user name, password and TFA token (if TFA is enabled).



Create Price f(x) configuration

Environment: QA

Partition: qa

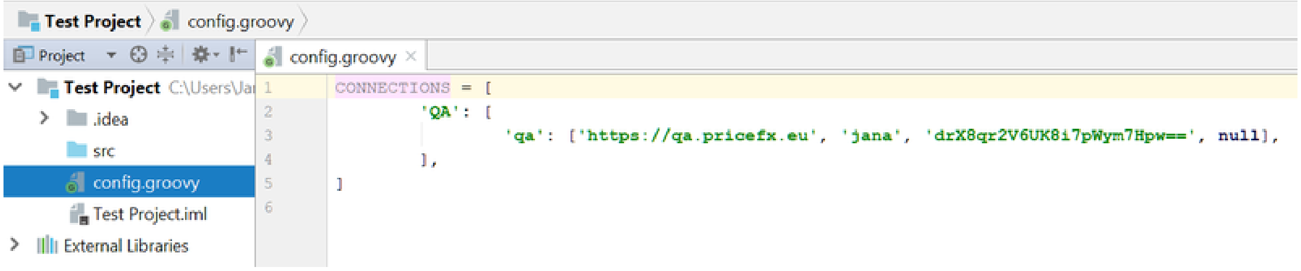
URL: https://qa.pricefx.eu

Username: jana

Password: •••

TFA Token:

3. Based on this data, a config file will be created. For example:



```
CONNECTIONS = [
        'QA': [
                'qa': ['https://qa.pricefx.eu', 'jana', 'drX8qr2V6UK8i7pWym7Hpw==', null],
        ],
]
```

4. If you need to add more environments and partitions, repeat the above steps.

Fetching Logics in Pricefx Studio in IDEA

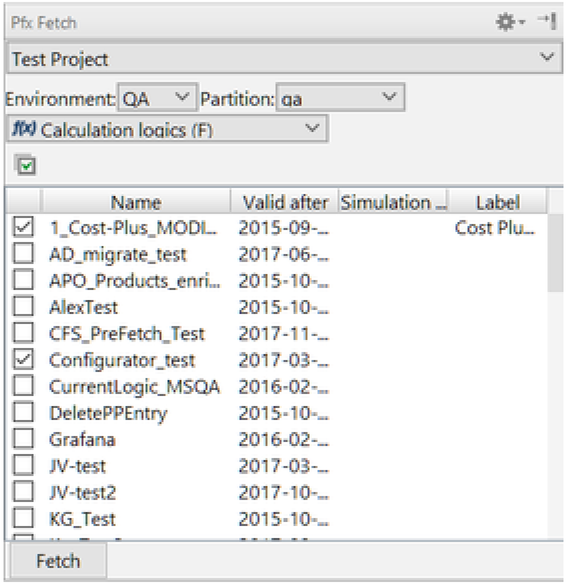
In Pricefx Studio you can work with all types of objects but for this tutorial we will pick logics as a typical example.

First you want to get the logics, so that that they are available in Pricefx Studio for editing and testing.

To fetch one or more logics:

1. Go to the Fetch panel. (If you do not see it, enable it in View > Tools Windows.)
2. Select the environment.
3. Select the partition.
4. Select 'Calculation logics (F)' as the type of object. The list of logics will be shown.
5. Select one or more logic to be fetched.
6. Click Fetch.

Pricefx Studio will fetch logics and write them to the project. For each logic there is one folder created.



Pfx Fetch

Test Project

Environment: QA Partition: qa

Calculation logics (F)

	Name	Valid after	Simulation ...	Label
☑	1_Cost-Plus_MODI...	2015-09-...		Cost Plu...
□	AD_migrate_test	2017-06-...		
□	APO_Products_enri...	2015-10-...		
□	AlexTest	2015-10-...		
□	CFS_PreFetch_Test	2017-11-...		
☑	Configurator_test	2017-03-...		
□	CurrentLogic_MSQA	2016-02-...		
□	DeletePPEntry	2015-10-...		
□	Grafana	2016-02-...		
□	JV-test	2017-03-...		
□	JV-test2	2017-10-...		
□	KG_Test	2015-10-...		

Editing Logics in Pricefx Studio in IDEA

- Logic File Format
- Editing an Element
- Adding a New Element
- Copying an Element
- Removing an Element

Logic File Format

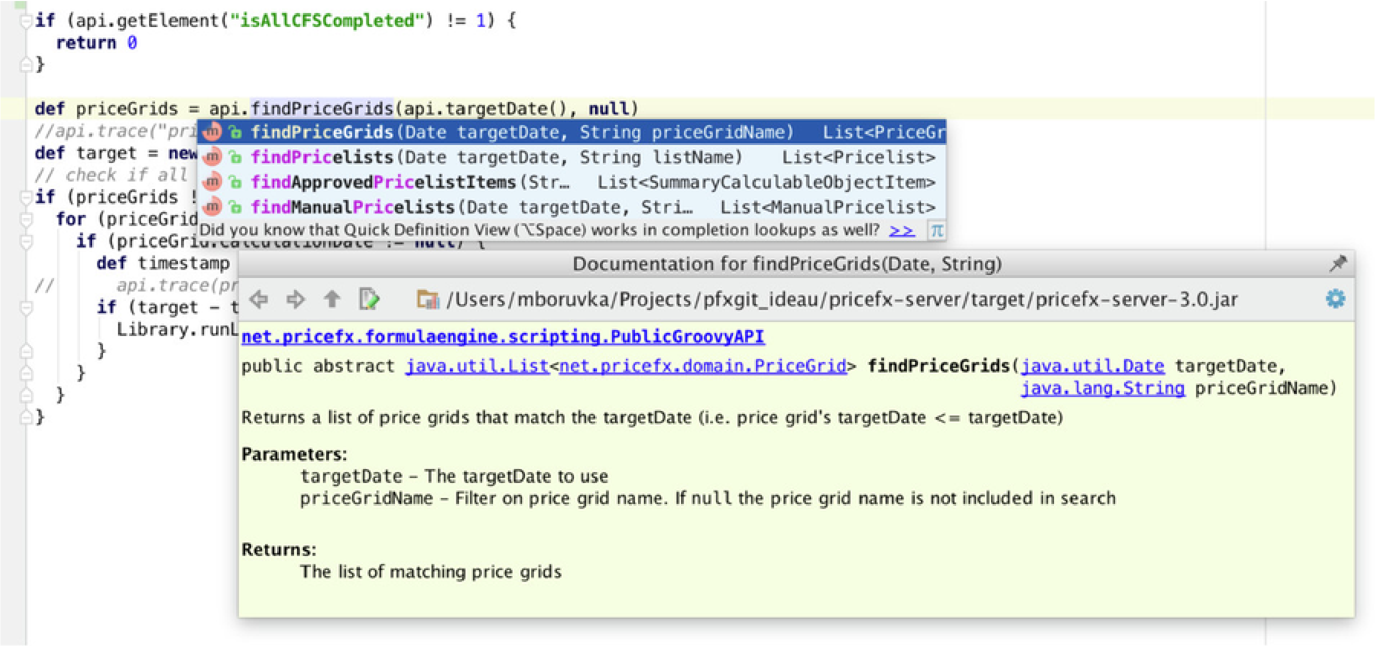
To edit logics, it is important to understand the structure of a project in Pricefx Studio. This is the same as when working with LM CLI client.

```
 my-project/                                      project folder
     CalculationLogic/                            folder storing
non-workflow logics
        Default_logic/                           logic folder
           elements/                            elements folder
              ApprovalReason.groovy            element file
              ItemGroup.formula                ...
              ProductGroup.formula
              ResultPrice.groovy
              ResultPriceReason.groovy
           logic.json                           logic file
        ProductInfo/
     WorkflowFormula/                             folder storing workflow
logics
     ProductExtension/                            folder storing Product
extension definitions
       ...
     config.groovy                                config holding PFX
connection
```

The logic file has one of the following extensions:

- *.formula (plain text files)
- *.groovy

Logics in the Groovy format get more support in the logic editor: JavaDoc for Pricefx public API and the code-complete assistant are available.



```
if (api.getElement("isAllCFSCompleted") != 1) {
  return 0
}

def priceGrids = api.findPriceGrids(api.targetDate(), null)
//api.trace("pri
def target = new
// check if all
if (priceGrids !
  for (priceGrid
    if (priceGrid.calculationDate != null) {
      def timestamp
//      api.trace(pr
      if (target - t
        Library.runL
      }
    }
  }
}
```

```
findPriceGrids(Date targetDate, String priceGridName)    List<PriceGr
findPricelists(Date targetDate, String listName)    List<Pricelist>
findApprovedPricelistItems(Str…    List<SummaryCalculableObjectItem>
findManualPricelists(Date targetDate, Stri…    List<ManualPricelist>
Did you know that Quick Definition View (⌥Space) works in completion lookups as well? >>
```

Documentation for findPriceGrids(Date, String)

/Users/mboruvka/Projects/pfxgit_ideau/pricefx-server/target/pricefx-server-3.0.jar

net.pricefx.formulaengine.scripting.PublicGroovyAPI

```
public abstract java.util.List<net.pricefx.domain.PriceGrid> findPriceGrids(java.util.Date targetDate,
                                                                            java.lang.String priceGridName)
```

Returns a list of price grids that match the targetDate (i.e. price grid's targetDate <= targetDate)

Parameters:

targetDate - The targetDate to use

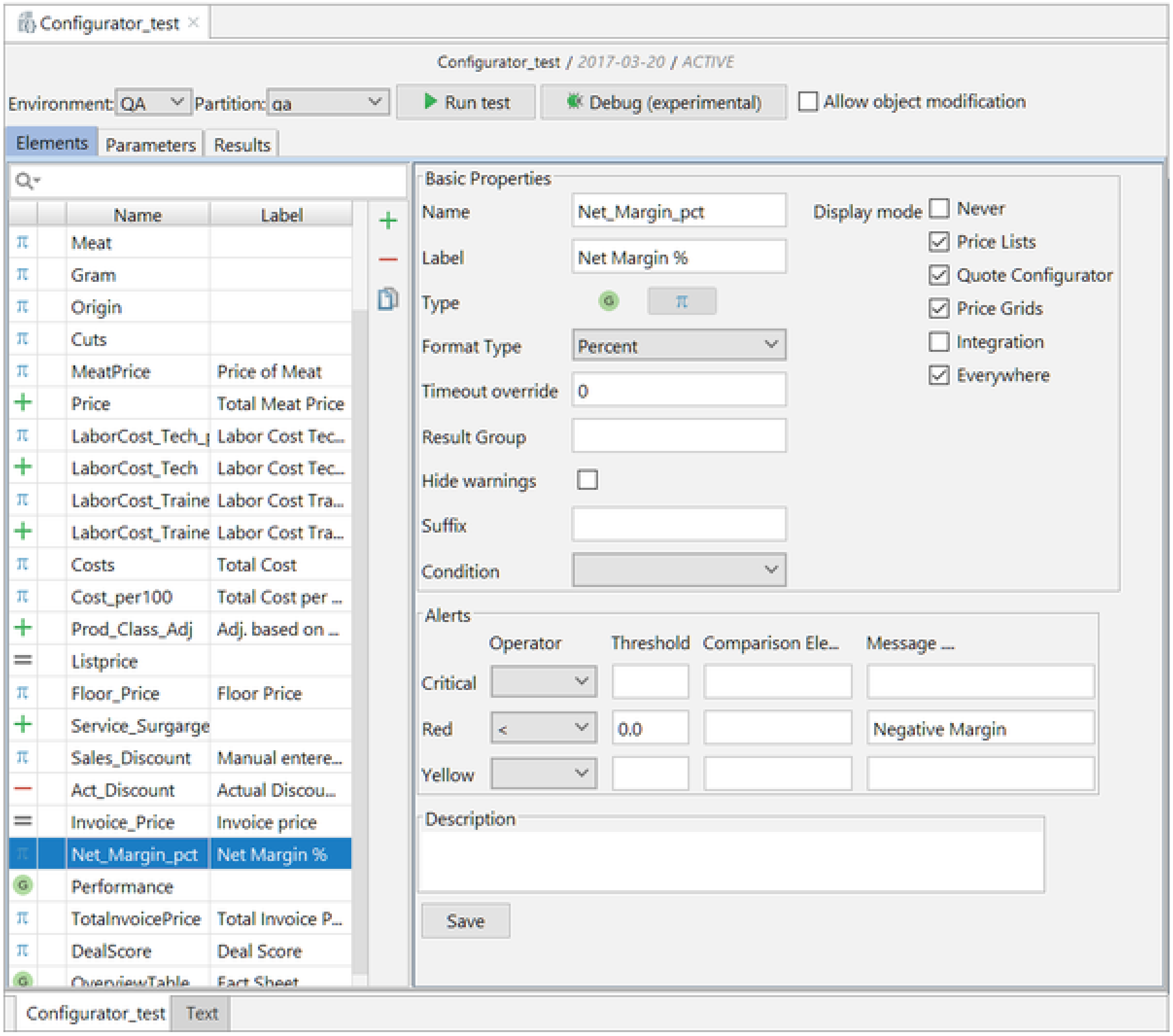
priceGridName - Filter on price grid name. If null the price grid name is not included in search

Returns:

The list of matching price grids

When you open the `logic.json` file, there are the following tabs available:

- **Elements** – Shows a list of elements and their Basic Properties, Alerts, Description and Display mode. Each element type has its own icon.
- **Parameters** – Allows you to define parameters for the test run. For details see Running a Logic Test in Pricefx Studio in IDEA.
- **Results** – Described in Running a Logic Test in Pricefx Studio in IDEA.



Configurator_test ×

Configurator_test / *2017-03-20* / *ACTIVE*

Environment: QA Partition: qa ▶ Run test Debug (experimental) □ Allow object modification

Elements | Parameters | Results

	Name	Label
π	Meat	
π	Gram	
π	Origin	
π	Cuts	
π	MeatPrice	Price of Meat
+	Price	Total Meat Price
π	LaborCost_Tech_|	Labor Cost Tec...
+	LaborCost_Tech	Labor Cost Tec...
π	LaborCost_Traine	Labor Cost Tra...
+	LaborCost_Traine	Labor Cost Tra...
π	Costs	Total Cost
π	Cost_per100	Total Cost per ...
+	Prod_Class_Adj	Adj. based on ...
=	Listprice	
π	Floor_Price	Floor Price
+	Service_Surgarge	
π	Sales_Discount	Manual entere...
−	Act_Discount	Actual Discou...
=	Invoice_Price	Invoice price
π	Net_Margin_pct	Net Margin %
G	Performance	
π	TotalnvoicePrice	Total Invoice P...
π	DealScore	Deal Score
G	OverviewTable	Fact Sheet

Basic Properties

Name: Net_Margin_pct

Label: Net Margin %

Type: G π

Format Type: Percent

Timeout override: 0

Result Group:

Hide warnings: □

Suffix:

Condition:

Display mode: □ Never ☑ Price Lists ☑ Quote Configurator ☑ Price Grids □ Integration ☑ Everywhere

Alerts

	Operator	Threshold	Comparison Ele...	Message ...
Critical				
Red	<	0.0		Negative Margin
Yellow				

Description

Save

Configurator_test | Text

Editing an Element

1. Select the element.
2. Modify its properties using the element panel on the right.
3. Click Save.

Adding a New Element

1. Click the Add + icon. A new element will be added to the list on the left.
2. Define its properties using the element panel on the right.
3. Click Save.

Copying an Element

1. Select the element.
2. Click the Copy 📋 icon.
3. Click Save.

Removing an Element

1. Select the element.
2. Click the Delete − icon.
3. Confirm the deletion.

If needed, you can do the same changes also in the code – on the Text tab.

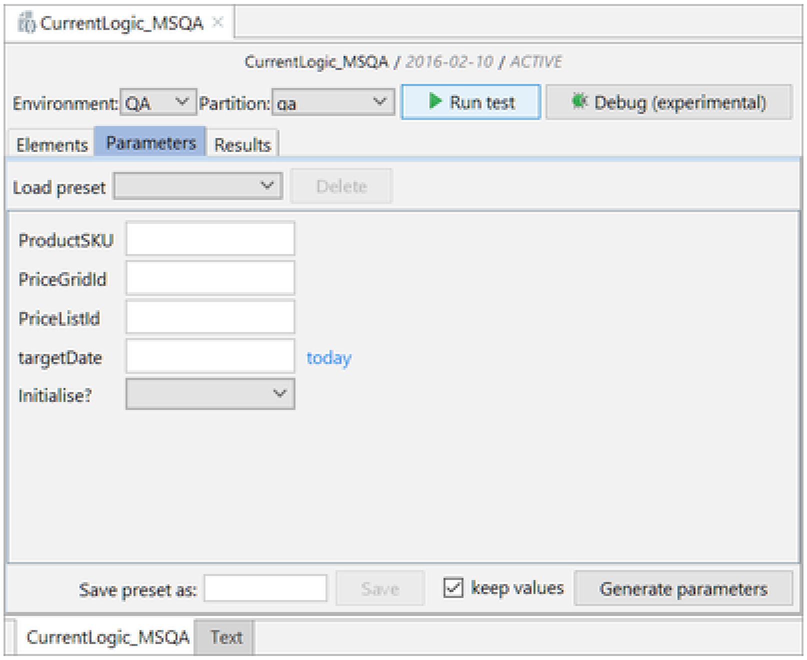
Running a Logic Test in Pricefx Studio in IDEA

- Defining Parameters for Test Run
- Running the Test
- Viewing Test Results

Defining Parameters for Test Run

To be able to run a logic test drive, you need to define the test parameters first.

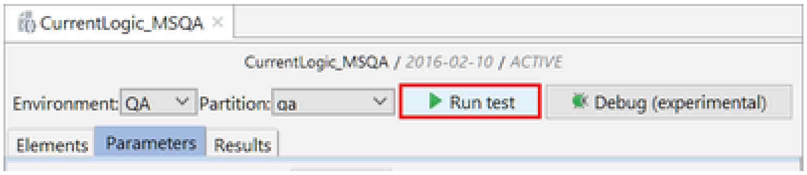
1. Open logic.json and go to the Parameters tab.
2. If this is the first run, click the 'Generate Parameters' button. This will populate the parameter fields for you to fill in.
   You need to include the "sku" parameter. The "targetDate" parameter is also needed, but if you omit it, it will be automatically set with today's date.
   ✓ You can then save the parameters for future use by using the options 'Save preset as' and 'Load preset'.
   If you generate the parameters with the option 'Keep values' enabled, those parameters which remained the same as in the last run keep the values they had filled in before.



Running the Test

After you have defined parameters, you can run the test:

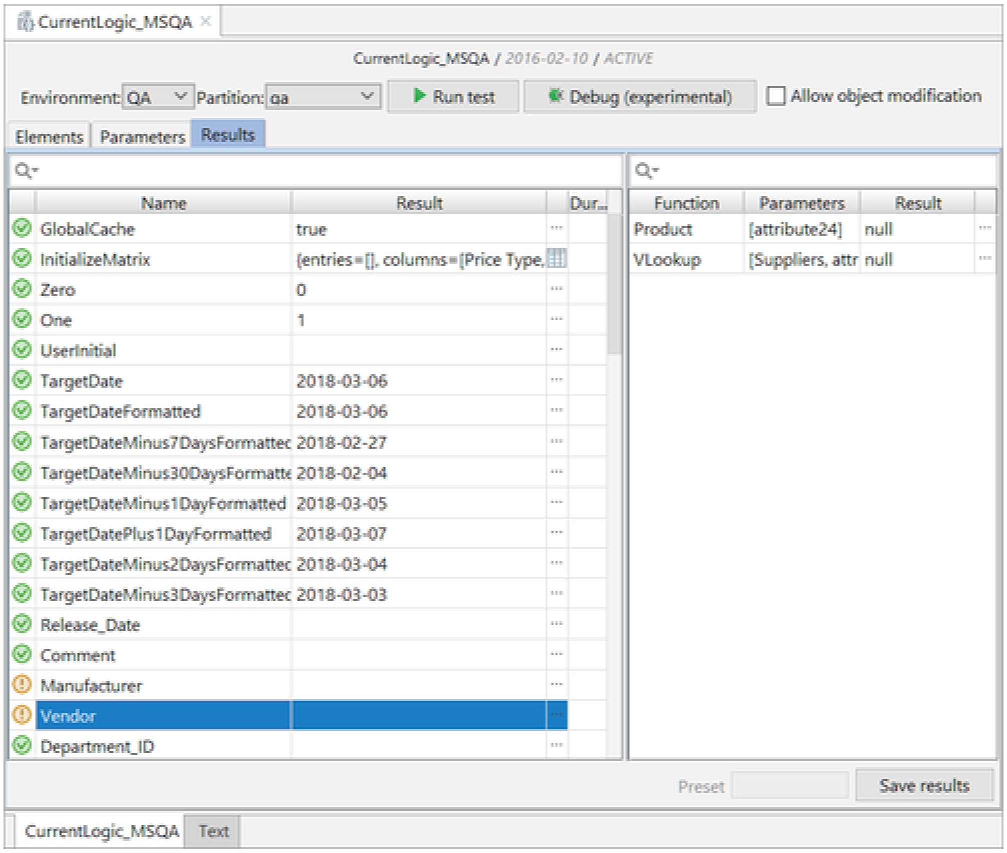
1. Open the logic.json file.
2. Select the environment and partition.
3. Click the Run Test button.



For details on the Debug function, see Debugging a Logic in Pricefx Studio in IDEA.

Viewing Test Results

When the test is completed, you get a list of executed elements and their values, traces and other details. The screenshot below illustrates the different types of results you can get, such as alerts, hyperlinks or errors.



Environment: QA Partition: qa □ Allow object modification

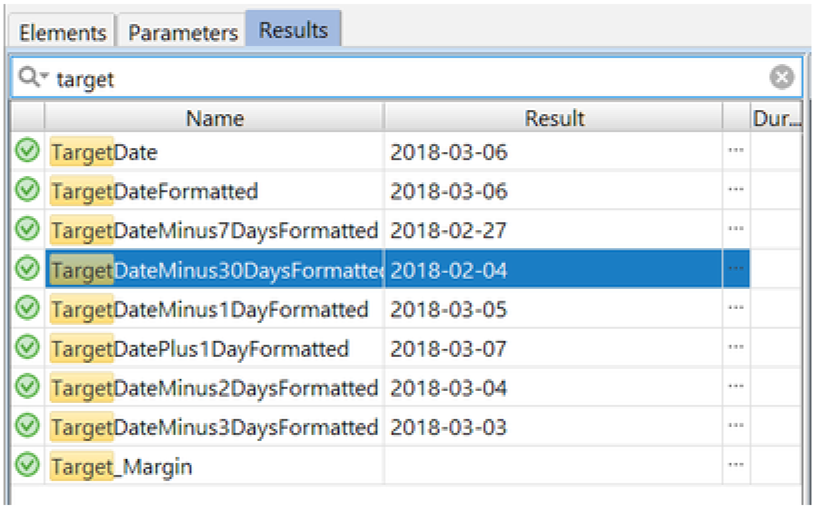
	Name	Result		Dur...
✓	GlobalCache	true	…	
✓	InitializeMatrix	{entries=[], columns=[Price Type,	▦	
✓	Zero	0	…	
✓	One	1	…	
✓	UserInitial		…	
✓	TargetDate	2018-03-06	…	
✓	TargetDateFormatted	2018-03-06	…	
✓	TargetDateMinus7DaysFormattec	2018-02-27	…	
✓	TargetDateMinus30DaysFormatte	2018-02-04	…	
✓	TargetDateMinus1DayFormatted	2018-03-05	…	
✓	TargetDatePlus1DayFormatted	2018-03-07	…	
✓	TargetDateMinus2DaysFormattec	2018-03-04	…	
✓	TargetDateMinus3DaysFormattec	2018-03-03	…	
✓	Release_Date		…	
✓	Comment		…	
!	Manufacturer		…	
!	Vendor		…	
✓	Department_ID		…	

Function	Parameters	Result	
Product	[attribute24]	null	…
VLookup	[Suppliers, attr	null	…

For each element, there are the following details displayed:

- Test status is indicated by an icon:
  - Green ✓ – successful run
  - Orange ! – warning
  - Red ⊖ – error
    Clicking on the icon opens the element file; in case of a parse or runtime error the cursor is placed at the line/column where it occurred.
- Result value
- Object browser … – Displays structure of the element results or element traces that are arrays and objects.

You can use the quick search field above the list to find a specific element:



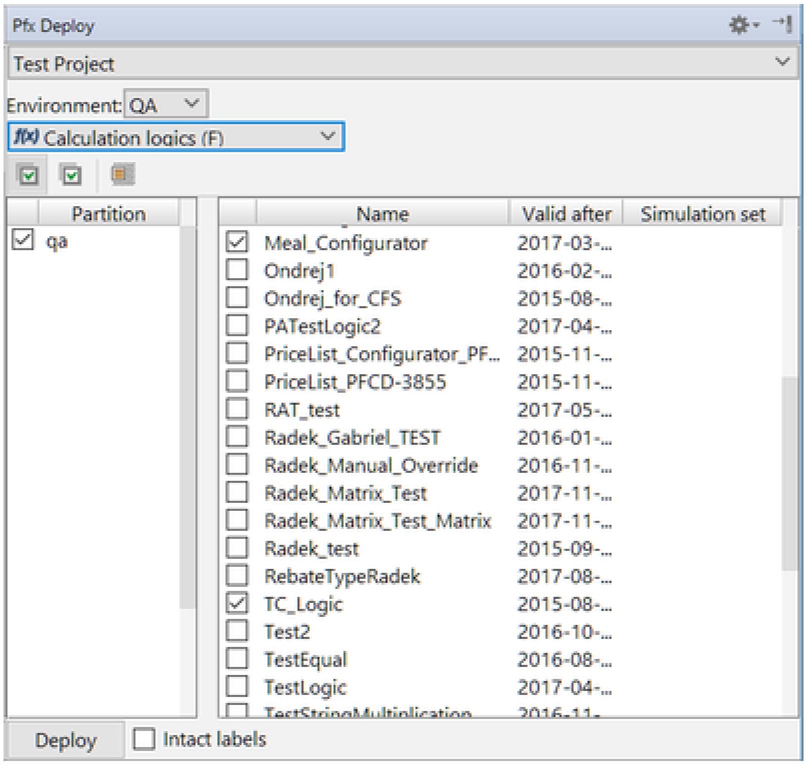
Search: target

	Name	Result		Dur...
✓	TargetDate	2018-03-06	…	
✓	TargetDateFormatted	2018-03-06	…	
✓	TargetDateMinus7DaysFormatted	2018-02-27	…	
✓	TargetDateMinus30DaysFormatte	2018-02-04	…	
✓	TargetDateMinus1DayFormatted	2018-03-05	…	
✓	TargetDatePlus1DayFormatted	2018-03-07	…	
✓	TargetDateMinus2DaysFormatted	2018-03-04	…	
✓	TargetDateMinus3DaysFormatted	2018-03-03	…	
✓	Target_Margin		…	

Deploying Logics in Pricefx Studio in IDEA

When you have finished editing the logics, you can deploy them, even to multiple partitions, at once.

1. Go to the Deploy panel. (If you do not see it, enable it in View > Tools Windows.)
2. Select the target environment.
3. Select one or more target partitions.
4. From the list of logics (which belong to the opened project) select one or more logics to be deployed.
5. Optionally, enable the 'Keep labels intact' option if the element labels should not be overwritten during deploy.
6. Click Deploy.



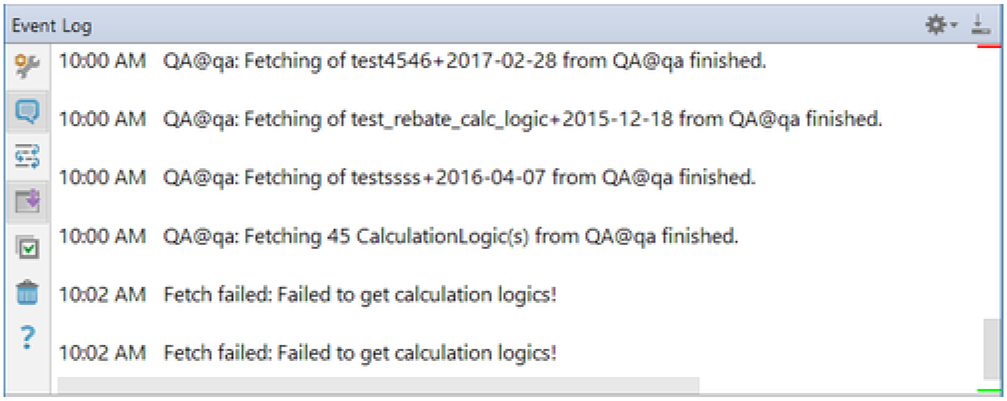
	Partition
☑	qa

	Name	Valid after	Simulation set
☑	Meal_Configurator	2017-03-...	
□	Ondrej1	2016-02-...	
□	Ondrej_for_CFS	2015-08-...	
□	PATestLogic2	2017-04-...	
□	PriceList_Configurator_PF...	2015-11-...	
□	PriceList_PFCD-3855	2015-11-...	
□	RAT_test	2017-05-...	
□	Radek_Gabriel_TEST	2016-01-...	
□	Radek_Manual_Override	2016-11-...	
□	Radek_Matrix_Test	2017-11-...	
□	Radek_Matrix_Test_Matrix	2017-11-...	
□	Radek_test	2015-09-...	
□	RebateTypeRadek	2017-08-...	
☑	TC_Logic	2015-08-...	
□	Test2	2016-10-...	
□	TestEqual	2016-08-...	
□	TestLogic	2017-04-...	
□	TestStringMultiplication	2016-11-	

Viewing Activity Log of Pricefx Studio in IDEA

In the bottom part of the window, you can view the Event Log with the log of performed actions. To display it, either click the bottom status bar or enable it in View > Tools Windows.

The potential logic related errors are displayed as hyperlinks that will open the related element file and move the cursor in the source code where the error occurred.



```
10:00 AM   QA@qa: Fetching of test4546+2017-02-28 from QA@qa finished.

10:00 AM   QA@qa: Fetching of test_rebate_calc_logic+2015-12-18 from QA@qa finished.

10:00 AM   QA@qa: Fetching of testssss+2016-04-07 from QA@qa finished.

10:00 AM   QA@qa: Fetching 45 CalculationLogic(s) from QA@qa finished.

10:02 AM   Fetch failed: Failed to get calculation logics!

10:02 AM   Fetch failed: Failed to get calculation logics!
```

Troubleshooting Pricefx Studio in IDEA

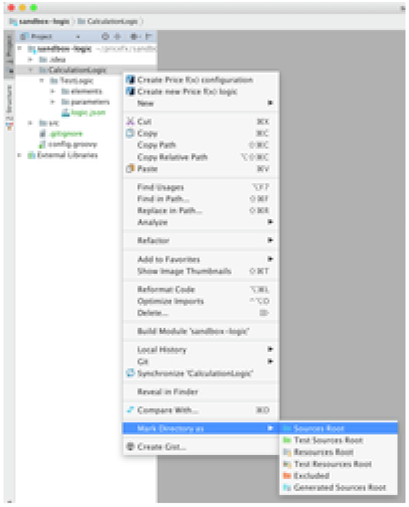
If you encounter issues while working with the IDEA plugin, please report them to support@pricefx.eu. If possible, attach the IDEA log file to the message. The log file is located at Help > Show Log in Files/Explorer/Finder; idea.log is the current log file.

- Why Auto-completion Is Not Working?
- Cannot Fetch Due to "Error: Empty Key"

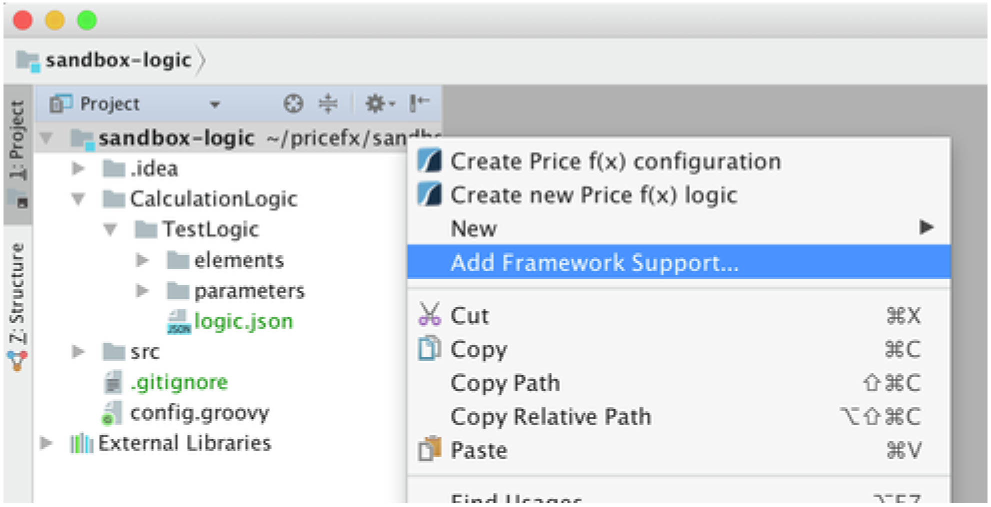
Why Auto-completion Is Not Working?

Check out the following:

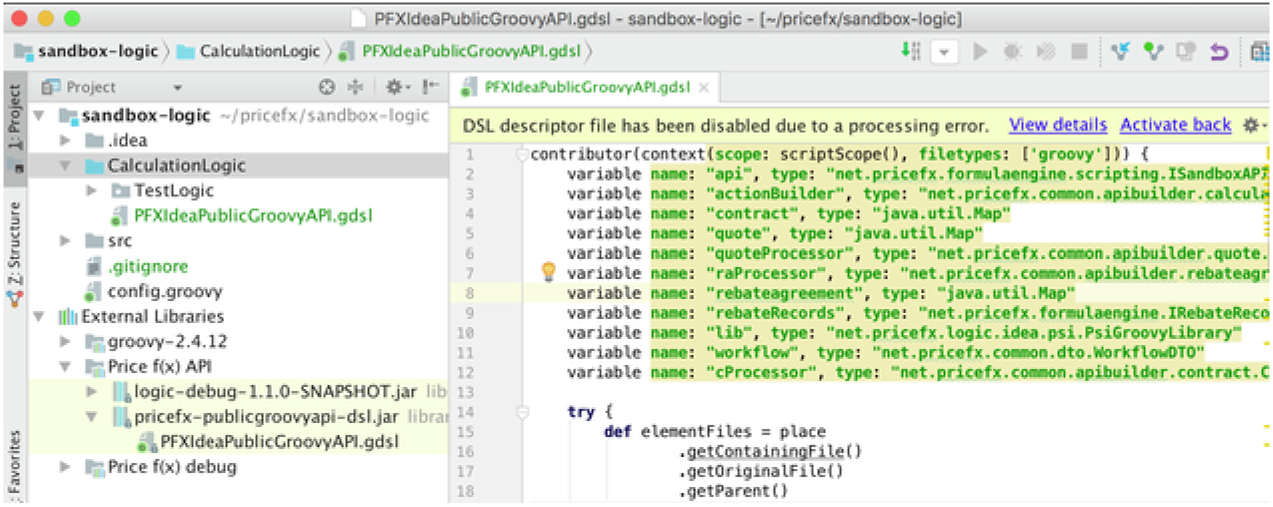
- Did you mark the folder with the logics as a Source folder in Module Settings?



- Did you add the Pricefx Framework support?



- Navigate to the External Libraries > Pricefx API > logic-debug-xxxx-xx-xx.jar > PFXIdeaPublicGroovyAPI.gdsl. If the auto-completion had probably crashed, then you have to reactivate it by clicking 'Activate back'.


PFXIdeaPublicGroovyAPI.gdsl - sandbox-logic - [~/pricefx/sandbox-logic]
sandbox-logic
CalculationLogic
PFXIdeaPublicGroovyAPI.gdsl
1: Project
Z: Structure
Favorites
Project
sandbox-logic ~/pricefx/sandbox-logic
.idea
CalculationLogic
TestLogic
PFXIdeaPublicGroovyAPI.gdsl
src
.gitignore
config.groovy
External Libraries
groovy-2.4.12
Price f(x) API
logic-debug-1.1.0-SNAPSHOT.jar lib
pricefx-publicgroovyapi-dsl.jar librar
PFXIdeaPublicGroovyAPI.gdsl
Price f(x) debug
PFXIdeaPublicGroovyAPI.gdsl
View details
Activate back

DSL descriptor file has been disabled due to a processing error.

```
contributor(context(scope: scriptScope(), filetypes: ['groovy'])) {
    variable name: "api", type: "net.pricefx.formulaengine.scripting.ISandboxAPI
    variable name: "actionBuilder", type: "net.pricefx.common.apibuilder.calcula
    variable name: "contract", type: "java.util.Map"
    variable name: "quote", type: "java.util.Map"
    variable name: "quoteProcessor", type: "net.pricefx.common.apibuilder.quote.
    variable name: "raProcessor", type: "net.pricefx.common.apibuilder.rebateagr
    variable name: "rebateagreement", type: "java.util.Map"
    variable name: "rebateRecords", type: "net.pricefx.formulaengine.IRebateReco
    variable name: "lib", type: "net.pricefx.logic.idea.psi.PsiGroovyLibrary"
    variable name: "workflow", type: "net.pricefx.common.dto.WorkflowDTO"
    variable name: "cProcessor", type: "net.pricefx.common.apibuilder.contract.C

    try {
        def elementFiles = place
                .getContainingFile()
                .getOriginalFile()
                .getParent()
```

- After doing "Activate Back", don't forget to purge the IntelliJ cache: File > Invalidate Caches / Restart > Invalidate and Restart.
    - This should force IntelliJ to rebuild caches for the API documentation.
- For cases when auto-completion does not work for referring elements either within the same logic or in libs and lib: if you use two

elements of the same name in the project, IDEA does not know which Class should be evaluated. Avoid having elements of the same name, so e.g. add a suffix "Utils" to the names of the Groovy library elements.

Cannot Fetch Due to "Error: Empty Key"

If you see "Error: Empty Key" in the fetch panel, then your config.groovy was created in a wrong way in the past. Just replace the part representing the two-factor authentication key in config.json – replace '' (empty String) with null. It is the last field of the array of connection details.

Debugging a Logic in Pricefx Studio in IDEA

How it Works

Formula elements are executed by a simplified formula engine in Pricefx Studio, similarly to the one running on the Pricefx backend server. The engine executes elements sequentially like ordinary Groovy scripts and the idea is that each of the Pricefx Groovy API is implemented as a separate stateless call of the backend server.

Limitations

- Due to the implementation, not all of the API functions are supported and it is not guaranteed that the results will be the identical to the standard formula test feature.
- Only logics with all elements of the Groovy type are supported.
- Mojito release is required.

Prerequisites

For the project in IDEA you need to:

- If you have some directories marked as "Source", remove such assignment in Module Settings > Modules > Sources (i.e. no directory should be marked as "Source").
- Set Output Path.
- Add Framework Support for Pricefx.

Debugging a Logic

1. Select parameters for the debug run as described here.
2. Put the breakpoint at the last line of the src/debug.groovy script. It will be created automatically the first time when you run the debugging.
3. Open the desired element file(s) and add breakpoint(s) on the lines where you want to stop the formula execution.
4. Click the 'Debug' button in the Logic editor. This starts the standard IDEA debugging of the debug.groovy script which will be automatically created in the src/ folder.
5. When the debugger stops here, add the variable watches.

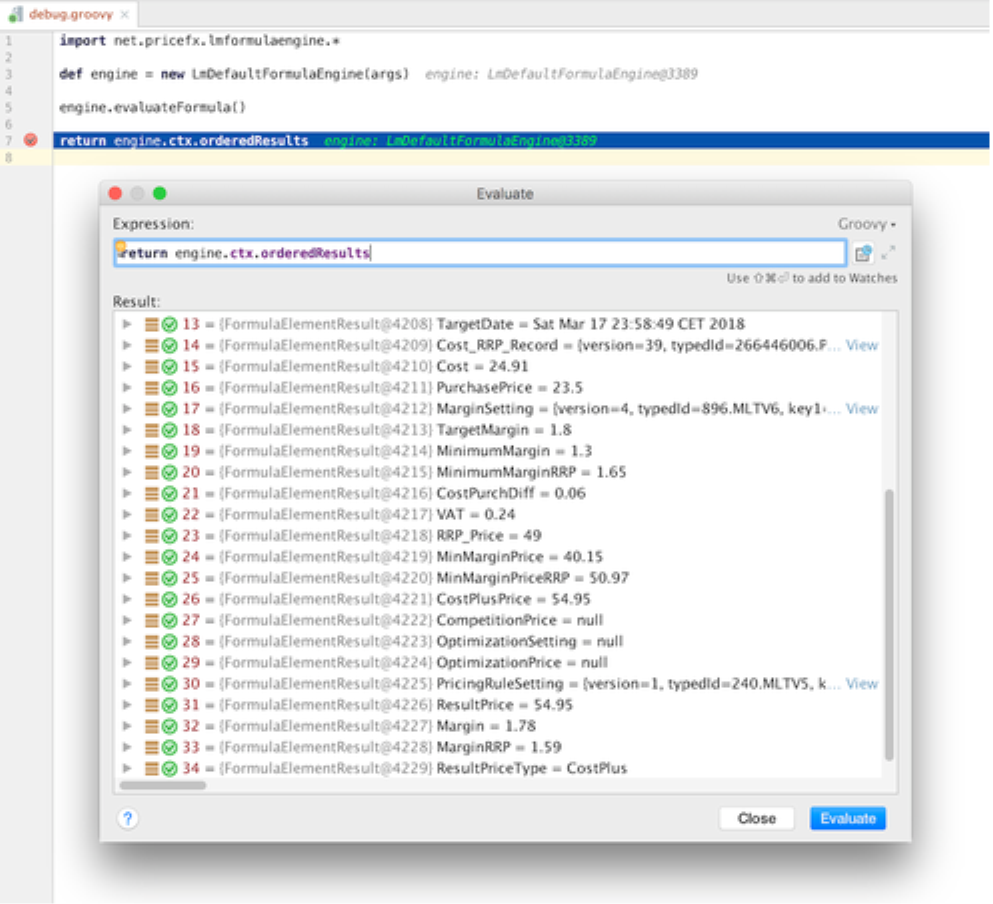
Hints for Variable Watches

Here are some tips for useful watches available from the element Groovy script:

- `api.ctx.orderedResultElements` – Shows all elements results.
- `api.ctx.resultElements.get("<SOME_ELEMENT_NAME>").elementResult` – Shows the result value of a certain element from your logic.
- `api.ctx.resultElements.get("<SOME_ELEMENT_NAME>")` – Shows details (such as api.traces, warnings etc.) of a certain element from your logic.
- `api.local` – Shows content of the `api.local` map.
- *`api.global`* – Shows content of the `api.global` map.

Analogically, some of the watches are available in debug.groovy as:

- `engine.ctx.orderedResultElements`
- `engine.ctx.resultElements.get("<SOME_ELEMENT_NAME>").elementResult`
- `engine.ctx.resultElements.get("<SOME_ELEMENT_NAME>")`



```
import net.pricefx.lmformulaengine.*

def engine = new LmDefaultFormulaEngine(args)  engine: LmDefaultFormulaEngine@3389

engine.evaluateFormula()

return engine.ctx.orderedResults  engine: LmDefaultFormulaEngine@3389
```

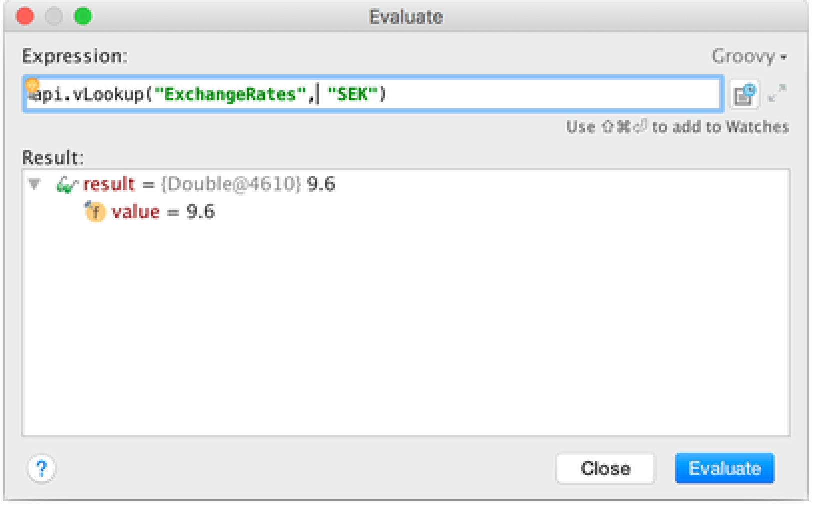
```
return engine.ctx.orderedResults
```

```
13 = {FormulaElementResult@4208} TargetDate = Sat Mar 17 23:58:49 CET 2018
14 = {FormulaElementResult@4209} Cost_RRP_Record = {version=39, typedId=266446006.F... View
15 = {FormulaElementResult@4210} Cost = 24.91
16 = {FormulaElementResult@4211} PurchasePrice = 23.5
17 = {FormulaElementResult@4212} MarginSetting = {version=4, typedId=896.MLTV6, key1-... View
18 = {FormulaElementResult@4213} TargetMargin = 1.8
19 = {FormulaElementResult@4214} MinimumMargin = 1.3
20 = {FormulaElementResult@4215} MinimumMarginRRP = 1.65
21 = {FormulaElementResult@4216} CostPurchDiff = 0.06
22 = {FormulaElementResult@4217} VAT = 0.24
23 = {FormulaElementResult@4218} RRP_Price = 49
24 = {FormulaElementResult@4219} MinMarginPrice = 40.15
25 = {FormulaElementResult@4220} MinMarginPriceRRP = 50.97
26 = {FormulaElementResult@4221} CostPlusPrice = 54.95
27 = {FormulaElementResult@4222} CompetitionPrice = null
28 = {FormulaElementResult@4223} OptimizationSetting = null
29 = {FormulaElementResult@4224} OptimizationPrice = null
30 = {FormulaElementResult@4225} PricingRuleSetting = {version=1, typedId=240.MLTV5, k... View
31 = {FormulaElementResult@4226} ResultPrice = 54.95
32 = {FormulaElementResult@4227} Margin = 1.78
33 = {FormulaElementResult@4228} MarginRRP = 1.59
34 = {FormulaElementResult@4229} ResultPriceType = CostPlus
```

Evaluate Expressions on the Fly

You can also evalute some api methods online e.g. `vLookup()` or `api.find()`:

1. Right click anywhere in the source code text of some element script and choose "Evaluate expression" (Alt + F8).
2. Put some expression there and press Enter.



```
api.vLookup("ExchangeRates", "SEK")
```

```
▼ result = {Double@4610} 9.6
    value = 9.6
```

Supported APIs

- Contexts: Product (P), PriceGrid items (PGI), PriceList (PLI)
- Groovy library (needs to be located in the same folder as the logic being debugged with a name '__LIBRARY__')
- `api.getElement()`
- `api.targetDate()`

- api.product(), api.productExtension(), api.productCompetition(), api.productXRef()
- api.currentItem()
- api.global, api.local maps, api.getGlobal(), api.putGlobal()
- api.find(), api.count()
- api.vLookup()
- api.trace()
- api.input(), api.userEntry(), api.stringUserEntry(), api.integerUserEntry(), api.decimalUserEntry(), api.booleanUserEntry(), api.dateUserEntry(), api.timeUserEntry(), api.dateTimeUserEntry(), api.option()
- api.newMatrix() (except transpose() method)
- api.criticalAlert(), api.redAlert(), api.yellowAlert()
- api.jsonEncode, api.jsonDecode(), api.jsonDecodeList()
- api.isSyntaxCheck() - returns always false
- api.isDebugMode() - returns always true
- api.isDistributedMode() - returns always false
- api.getId()
- api.calendar()
- api.random(), api.uuid()
- api.getMaxFindResultsLimit()
- api.formatNumber()
- api.abortCalculation()
- api.getTimeZone()
- api.filter, api.filterFromMap()
- api.user(), api.roles(), api.isUserInGroup()
- api.log(), api.logWarn(), api.logInfo()
- api.pricelist()
- api.pricelistItem()
- api.getPricelistItem()
- api.customer()
- api.otherProduct()
- api.productCompetition()
- api.productXRef()
- api.otherCustomer()
- api.getCalculableLineItemCollection()
- api.bom()
- api.bomList()
- api.findLookupTableValues()
- api.getBoundPartitionNames()
- api.contextByLabel()
- api.findDataLoad()
- api.findRebateRecordLoad()
- api.attributedResult()

- api.httpCall()
- api.boundCall()

Supported APIs with no effect in debug mode:

- api.add()
- api.addOrUpdate()
- api.update()
- api.delete()
- api.sendEmail()
- api.createNewRevision()

What Is on the Roadmap

- support for the 'libs'
- Debugging of a multiple P, PGI or PLI records => support for 1st and 2nd pass
    - api.markItemDirty()
    - api.markItemClean()
    - api.isSecondPass()
- Contexts related:
    - api.currentContext()
    - api.getIterationNumber()
    - api.switchContext()
    - api.switchSKUContext()
    - api.resetSKUContextSwitch()
- Decorations and attributed elements

What Will Not Be Supported

- Other than Groovy elements
- API functions which use other than basic Java types as input/output parameters (DatamartContext, ContextParameter)
- PA logics and datamart context related

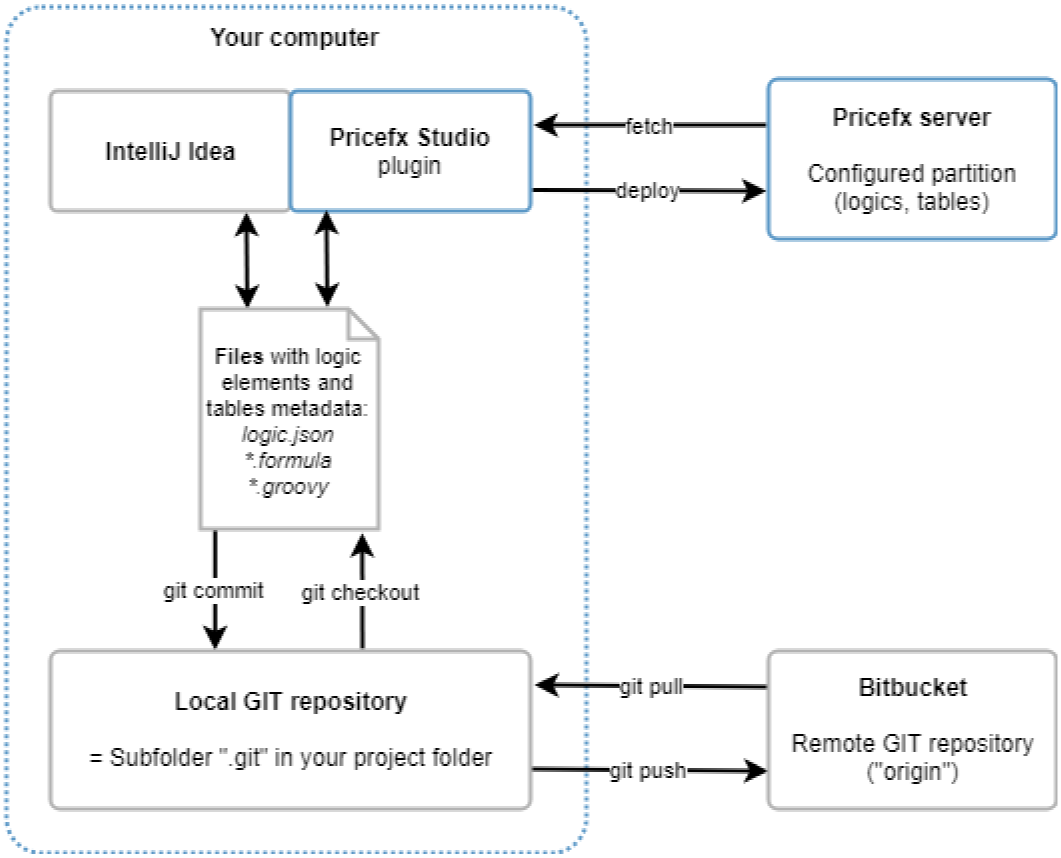
Best Practises for Logic Development with Debug Support

- Do not use the non-Groovy type of elements.
- Put if-else statements on multiple lines so you can place breakpoints and step in.

Keyboard Shortcuts in Studio

Windows	Mac	Where	Action
CTRL + SHIFT + L	+ SHIFT + L	Anywhere	Opens a logic
CTRL + SHIFT + N	+ SHIFT + O	Anywhere	Opens an element
Alt + 8	+ 8	Anywhere	Opens/closes the Fetch tool window
Alt + 0	+ 0	Anywhere	Opens/closes the Deploy tool window
hold CTRL and click	hold and click	Anywhere in the code on api.getElement("here")	Jumps in the referred element
CTRL + J	F1	In the code on a Groovy API function	Displays the offline documentation (Javadoc) of a Groovy API function
CTRL + P	+ P	In the code in a Groovy API function parentheses	Displays the Groovy API function parameters
CTRL + SHIFT + F	+ SHIFT + F	On a folder in a project tree (e.g. CalculationLogic)	Searches for text across all element sources
CTRL + /	+ /	Anywhere in the code	Comments out a line in the code
CTRL + ALT + /	+ ALT + /	Anywhere in the code	Comments out a block of code
SHIFT + F6	SHIFT + F6	In the code on a variable occurrence	Refactors (renames) the Groovy variable
Alt + F5	Alt + F5	logic.json editor or the element source code editor (logic.json must be opened)	Runs the logic test
CTRL + ENTER	+ ENTER	Groovy console tab in the logic.json window PA SQL query console	Evaluates the expression / Run SQL
Ctrl + A	Ctrl + A	Fetch or Deploy tool windows	Selects all logics
Tab	Tab	logic.json editor parameters	Jumps over the parameters
pfxfind<ENTER>	pfxfind<ENTER>	Anywhere in the code	Generates a code snippet for the typical api.find() call with filters
/**<ENTER>	/**<ENTER>	Anywhere in the code above the function declaration	Generate a Javadoc template of a Groovy function
CTRL + L	+ ALT + L	Anywhere in the code	Reformats the Groovy code according to a standard Java code style convention

GIT and Pricefx Studio


Your computer
IntelliJ Idea
Pricefx Studio plugin
fetch
deploy
Pricefx server
Configured partition (logics, tables)
Files with logic elements and tables metadata: *logic.json* **.formula* **.groovy*
git commit
git checkout
Local GIT repository
= Subfolder ".git" in your project folder
git pull
git push
Bitbucket
Remote GIT repository ("origin")

IntelliJ IDEA Shortcuts and Helpful Hints

General Troubleshooting

If IDEA is behaving strangely, there is a magic fix-all solution which will fix most issues.

1. Invalidate caches
   File > Invalidate Cache / Restart > Invalidate
2. Close IntelliJ.
   File > Exit
3. Delete IntelliJ config files in project.
   Browse to project folder
   Find .iml file in project root and DELETE it.
   Find .idea folder in project root and DELETE it.
4. Reopen Project in IDEA.
5. Invalidate and restart IDEA.
   File > Invalidate Cache / Restart > Invalidate & Restart

IDEA Default Shortcuts

Win	Mac	Result
Ctrl + P	Cmd + P	View Parameter Info (View all parameter overloads of method)
Ctrl + Q	Ctrl + J	View Quick Documentation (View method sig / return type etc.)
Ctrl + Alt + L		Auto-Format Selected Code

Code Reference Info:
https://www.jetbrains.com/help/idea/viewing-reference-information.html

Used Free and Open Source Software

This document summarizes free and open source software (FOSS) components used in Pricefx Studio. The licenses referenced by individual components are provided at the end of this document.

Used FOSS Components

The following free and open source software components were used to develop Pricefx Studio. For each component the table summarizes its category, name, version, copyright notice and references to the project, manifest license, POM license, embedded license files (as applicable). The source code, where required, can be downloaded from this page.

Group	ch.qos.logback
Name	logback-classic
Version	1.1.3
Source Code	Download
Copyright Notice	Copyright © 2019 QOS.ch
Manifest Project URL	http://www.qos.ch
Manifest License URL	http://www.eclipse.org/legal/epl-v10.html https://www.gnu.org/licenses/old-licenses/lgpl-2.1.html
POM License	Eclipse Public License 1.0 – http://www.eclipse.org/legal/epl-v10.html GNU Lesser General Public License – https://www.gnu.org/licenses/old-licenses/lgpl-2.1.html

Group	ch.qos.logback
Name	logback-core
Version	1.1.3
Source Code	Download
Copyright Notice	Copyright © 2019 QOS.ch
Manifest Project URL	http://www.qos.ch
Manifest License URL	http://www.eclipse.org/legal/epl-v10.html https://www.gnu.org/licenses/old-licenses/lgpl-2.1.html
POM License	Eclipse Public License 1.0 – http://www.eclipse.org/legal/epl-v10.html GNU Lesser General Public License – https://www.gnu.org/licenses/old-licenses/lgpl-2.1.html

Group	com.brsanthu
Name	migbase64
Version	2.2
Copyright Notice	Copyright © 2004, Mikael Grev, MiG InfoCom AB
Project URL	http://sourceforge.net/projects/migbase64/
Manifest License URL	http://en.wikipedia.org/wiki/BSD_licenses
POM License	Prior BSD License – http://en.wikipedia.org/wiki/BSD_licenses

Group	com.fasterxml.jackson.core
Name	jackson-annotations

Version	2.8.4
Copyright Notice	Copyright © 2009 FasterXML, LLC
Project URL	http://github.com/FasterXML/jackson
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	jackson-annotations-2.5.4.jar/META-INF/LICENSE

Group	com.fasterxml.jackson.core
Name	jackson-core
Version	2.8.4
Copyright Notice	Copyright © 2009 FasterXML, LLC
Project URL	https://github.com/FasterXML/jackson
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	jackson-core-2.5.4.jar/META-INF/LICENSE jackson-core-2.5.4.jar/META-INF/NOTICE

Group	com.fasterxml.jackson.core
Name	jackson-databind
Version	2.8.4
Copyright Notice	Copyright © 2009 FasterXML, LLC
Project URL	http://github.com/FasterXML/jackson
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	jackson-databind-2.5.4.jar/META-INF/LICENSE jackson-databind-2.5.4.jar/META-INF/NOTICE

Group	com.fasterxml.jackson.datatype
Name	jackson-datatype-joda
Version	2.1.5
Copyright Notice	Copyright © 2009 FasterXML, LLC
Project URL	http://wiki.fasterxml.com/JacksonModuleJoda
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	com.fasterxml.jackson.jaxrs
Name	jackson-jaxrs-base
Version	2.5.4
Copyright Notice	Copyright © 2009 FasterXML, LLC
Manifest Project URL	https://github.com/FasterXML
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	com.fasterxml.jackson.jaxrs
Name	jackson-jaxrs-providers
Version	2.5.4
Copyright Notice	Copyright © 2009 FasterXML, LLC
Manifest Project URL	https://github.com/FasterXML
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	jackson-jaxrs-json-provider-2.5.4.jar/META-INF/LICENSE jackson-jaxrs-json-provider-2.5.4.jar/META-INF/NOTICE

Group	com.fasterxml.jackson.module
Name	jackson-module-jaxb-annotations
Version	2.5.4
Copyright Notice	Copyright © 2009 FasterXML, LLC
Project URL	https://github.com/FasterXML/jackson-module-jaxb-annotations
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	jackson-module-jaxb-annotations-2.5.4.jar/META-INF/LICENSE jackson-module-jaxb-annotations-2.5.4.jar/META-INF/NOTICE

Group	com.google.guava
Name	guava
Version	19.0
Copyright Notice	Copyright © 2019 The Guava Authors
Manifest Project URL	https://guava-libraries.googlecode.com/
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	com.j256.two-factor-auth
Name	two-factor-auth
Version	1.0
Copyright Notice	Copyright © 2019 OATH
Manifest Project URL	https://github.com/j256/two-factor-auth
Manifest License URL	https://opensource.org/licenses/ISC
POM License	ISC License – https://opensource.org/licenses/ISC

Group	com.squareup.okhttp3
Name	logging-interceptor
Version	3.8.0
Copyright Notice	Copyright © 2019 Square
Manifest Project URL	https://mvnrepository.com/artifact/com.squareup.okhttp3/logging-interceptor/3.8.0
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	com.squareup.okhttp3
Name	okhttp
Version	3.8.0
Copyright Notice	Copyright © 2019 Square
Manifest Project URL	https://github.com/square/okhttp
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	com.squareup.okio
Name	okio
Version	1.13.0
Copyright Notice	Copyright © 2019 Square
Manifest Project URL	https://github.com/square/okio
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	com.googlecode.java-diff-utils
Name	diffutils

Version	1.2.1
Copyright Notice	Copyright © 2010 Free Software Foundation, Inc.
POM Project URL	http://code.google.com/p/java-diff-utils/
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	com.jetbrains
Name	idea
Version	2016.3
Copyright Notice	Copyrigh © 2000—2019 JetBrains s.r.o.
POM Project URL	http://logging.apache.org/log4j/1.2/
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	commons-codec-1.9.jar/META-INF/LICENSE.txt commons-codec-1.9.jar/META-INF/NOTICE.txt

Group	com.sun.jersey
Name	jersey-bundle
Version	1.19
Source Code	Download
Copyright Notice	Copyright © 2010-2019 Oracle Corporation
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	com.sun.jersey
Name	jersey-core
Version	1.19
Source Code	Download
Copyright Notice	Copyright © 2010-2019 Oracle Corporation
Manifest Project URL	http://www.oracle.com
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	com.sun.jersey.contribs
Name	jersey-multipart
Version	1.19

Source Code	Download
Copyright Notice	Copyright © 2010-2019 Oracle Corporation
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	commons-beanutils
Name	commons-beanutils
Version	1.9.2
Copyright Notice	Copyright © 2000-2016 The Apache Software Foundation
Project URL	http://commons.apache.org/proper/commons-beanutils/
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	commons-beanutils-1.9.2.jar/META-INF/LICENSE.txt commons-beanutils-1.9.2.jar/META-INF/NOTICE.txt

Group	commons-codec
Name	commons-codec
Version	1.9
Copyright Notice	Copyright © 2002-2014 The Apache Software Foundation
Project URL	http://commons.apache.org/proper/commons-codec/
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	commons-codec-1.9.jar/META-INF/LICENSE.txt commons-codec-1.9.jar/META-INF/NOTICE.txt

Group	commons-collections
Name	commons-collections
Version	3.2.1
Copyright Notice	Copyright © 2001-2019 The Apache Software Foundation
Project URL	http://commons.apache.org/collections/
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	commons-collections-3.2.1.jar/META-INF/LICENSE.txt commons-collections-3.2.1.jar/META-INF/NOTICE.txt

Group	commons-lang
Name	commons-lang
Version	2.6
Copyright Notice	Copyright © 2001-2019 The Apache Software Foundation
Project URL	http://commons.apache.org/lang/
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	commons-lang-2.6.jar/META-INF/LICENSE.txt commons-lang-2.6.jar/META-INF/NOTICE.txt

Group	commons-logging
Name	commons-logging
Version	1.1.1
Copyright Notice	Copyright © 2001-2014 The Apache Software Foundation
POM Project URL	http://commons.apache.org/logging
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	commons-logging-1.1.1.jar/META-INF/LICENSE commons-logging-1.1.1.jar/META-INF/NOTICE

Group	de.vandermeer
Name	asciilist
Version	0.0.3
Copyright Notice	Copyright © 2016 Sven van der Meer vdmeer.sven@mykolab.com
POM Project URL	https://github.com/vdmeer/asciilist
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	asciilist-0.0.3.jar/LICENSE asciilist-0.0.3.jar/README.asciidoc asciilist-0.0.3.jar/README.html

Group	de.vandermeer
Name	asciitable
Version	0.2.5
Copyright Notice	Copyright © 2019 Sven van der Meer vdmeer.sven@mykolab.com
POM Project URL	https://github.com/vdmeer/asciitable
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	asciitable-0.2.5.jar/LICENSE asciitable-0.2.5.jar/README.asciidoc asciitable-0.2.5.jar/README.html

Group	io.swagger
Name	swagger-annotations
Version	1.5.0
Copyright Notice	Copyright © 2019 SmartBear Software
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.html
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	javax.activation
Name	activation
Version	1.1
Source Code	Download
Copyright Notice	Copyright © 1996-2015 Oracle and/or its affiliates
POM Project URL	http://java.sun.com/products/javabeans/jaf/index.jsp
POM License	CDDL 1.0 – https://glassfish.dev.java.net/public/CDDLv1.0.html
Embedded License Files	activation-1.1.jar/META-INF/LICENSE.txt

Group	javax.annotation
Name	javax.annotation-api
Version	1.2
Source Code	Download
Copyright Notice	Copyright © 2005-2010 Oracle and/or its affiliates
Manifest Project URL	https://glassfish.java.net
Manifest License URL	https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html
POM Project URL	http://jcp.org/en/jsr/detail?id=250
POM License	CDDL 1.0 – https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html
Embedded License Files	javax.annotation-api-1.2.jar/META-INF/LICENSE.txt

Group	javax.inject
Name	javax.inject
Version	1
Copyright Notice	Copyright © 2009 The JSR-330 Expert Group
POM Project URL	http://code.google.com/p/atinject/
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	javax.mail
Name	mail
Version	1.4
Source Code	Download
Copyright Notice	Copyright © 1996-2015 Oracle and/or its affiliates
POM Project URL	https://glassfish.dev.java.net/javaee5/mail/
POM License	CDDL 1.0 – https://glassfish.dev.java.net/public/CDDLv1.0.html
Embedded License Files	mail-1.4.jar/META-INF/LICENSE.txt

Group	javax.ws.rs
Name	javax.ws.rs-api
Version	2.0.1
Source Code	Download
Copyright Notice	Copyright © 2012-2015 Oracle and/or its affiliates
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM Project URL	http://jax-rs-spec.java.net
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	javax.ws.rs
Name	jsr311-api
Version	1.1.1
Source Code	Download
Copyright Notice	Copyright © 2010 Oracle and/or its affiliates
Manifest Project URL	http://www.sun.com/
Manifest License URL	http://www.opensource.org/licenses/cddl1.php
POM Project URL	https://jsr311.dev.java.net
POM License	CDDL – http://www.opensource.org/licenses/cddl1.php

Group	joda-time
Name	joda-time
Version	2.3
Copyright Notice	Copyright © 2002-2019 Joda.org
Project URL	http://www.joda.org/joda-time/

Manifest License	Apache 2.0 (Not packaged)
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	joda-time-2.3.jar/META-INF/LICENSE.txt joda-time-2.3.jar/META-INF/NOTICE.txt

Group	net.java.dev.rome
Name	rome
Version	1.0.0
Copyright Notice	Copyright © 2011 The ROME Team
Manifest Project URL	https://rometools.github.io/rome
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM Project URL	https://rome.dev.java.net/
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	net.pricefx.automation
Name	pricefx.automation
Version	1.0.2-SNAPSHOT

Group	net.pricefx.integration
Name	pricefx-api
Version	1.0.2-SNAPSHOT

Group	net.pricefx.integration
Name	pricefx-api-client
Version	1.0.2-SNAPSHOT

Group	net.pricefx.logic
Name	logic
Version	0.15-SNAPSHOT

Group	net.pricefx.logic
Name	logic-cli
Version	0.15-SNAPSHOT

Group	net.pricefx.logic
Name	logic-core
Version	0.15-SNAPSHOT

Group	net.sourceforge.argparse4j
Name	argparse4j
Version	0.7.0
Copyright Notice	Copyright © 2011, 2015 Tatsuhiro Tsujikawa
POM Project URL	http://argparse4j.github.io
POM License	MIT – https://raw.github.com/tatsuhiro-t/argparse4j/master/LICENSE.txt

Group	org.apache.commons
Name	commons-lang3
Version	3.4
Copyright Notice	Copyright © 2001-2019 The Apache Software Foundation
Project URL	http://commons.apache.org/proper/commons-lang/
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	commons-lang3-3.4.jar/META-INF/LICENSE.txt commons-lang3-3.4.jar/META-INF/NOTICE.txt

Group	org.slf4j
Name	jcl-over-slf4j
Version	1.7.21
Copyright Notice	Copyright © 2004-2019 QOS.ch
Manifest Project URL	https://guava-libraries.googlecode.com/
Manifest License URL	http://www.opensource.org/licenses/mit-license.php
POM License	MIT – http://www.opensource.org/licenses/mit-license.php

Group	org.aspectj
Name	aspectjrt
Version	1.8.5
Source Code	Download
Copyright Notice	Copyright © 2019 The Eclipse Foundation

POM Project URL	https://eclipse.org/aspectj/
POM License	Eclipse Public License 1.0 – http://www.eclipse.org/legal/epl-v10.html

Group	org.aspectj
Name	aspectjtools
Version	1.8.5
Source Code	Download
Copyright Notice	Copyright © 2019 The Eclipse Foundation
POM Project URL	https://eclipse.org/aspectj/
POM License	Eclipse Public License 1.0 – http://www.eclipse.org/legal/epl-v10.html
Embedded License Files	aspectjtools-1.8.5.jar/about_files/NOTICE.txt

Group	org.codehaus.groovy
Name	groovy-all
Version	2.4.6
Copyright Notice	Copyright © 2003-2019 The Apache Groovy Project
POM Project URL	http://groovy-lang.org
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	groovy-all-2.4.6.jar/META-INF/LICENSE groovy-all-2.4.6.jar/META-INF/NOTICE

Group	org.codehaus.jettison
Name	jettison
Version	1.3.7
Copyright Notice	Copyright © 2006 Envoi Solutions LLC
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	jettison-1.3.7.jar/META-INF/LICENSE

Group	org.glassfish.hk2
Name	hk2-api
Version	2.4.0-b31
Source Code	Download
Copyright Notice	Copyright © 2015 Oracle, Inc.
Manifest Project URL	http://www.oracle.com

Manifest License URL	https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.glassfish.hk2
Name	hk2-locator
Version	2.4.0-b31
Source Code	Download
Copyright Notice	Copyright © 2015 Oracle, Inc.
Manifest Project URL	http://www.oracle.com
Manifest License URL	https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.glassfish.hk2
Name	hk2-utils
Version	2.4.0-b31
Source Code	Download
Copyright Notice	Copyright © 2015 Oracle, Inc.
Manifest Project URL	http://www.oracle.com
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.glassfish.hk2
Name	osgi-resource-locator
Version	1.0.1
Source Code	Download
Copyright Notice	Copyright © 2015 Oracle, Inc.
Manifest Project URL	https://glassfish.dev.java.net
Manifest License URL	https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html
POM License	CDDL – https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html

Group	org.glassfish.hk2.external
Name	aopalliance-repackaged
Version	2.4.0-b31
Source Code	Download
Copyright Notice	Copyright © 2015 Oracle, Inc.

Manifest Project URL	http://www.oracle.com
Manifest License URL	https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html
POM License	CDDL – https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html

Group	org.glassfish.hk2.external
Name	javax.inject
Version	2.4.0-b31
Source Code	Download
Copyright Notice	Copyright © 2015 Oracle, Inc.
Manifest Project URL	http://www.oracle.com
Manifest License URL	https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html
POM License	CDDL – https://glassfish.java.net/nonav/public/CDDL+GPL_1_1.html

Group	org.glassfish.jersey.bundles.repackaged
Name	jersey-guava
Version	2.22.1
Source Code	Download
Copyright Notice	Copyright © 2014 Oracle, Inc.
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.glassfish.jersey.core
Name	jersey-client
Version	2.22
Source Code	Download
Copyright Notice	Copyright © 2012 Oracle, Inc.
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.glassfish.jersey.core
Name	jersey-common
Version	2.22.1

Source Code	Download
Copyright Notice	Copyright © 2012 Oracle, Inc.
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.glassfish.jersey.ext
Name	jersey-entity-filtering
Version	2.22.1
Source Code	Download
Copyright Notice	Copyright © 2013 Oracle, Inc.
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.glassfish.jersey.media
Name	jersey-media-json-jackson
Version	2.22.1
Source Code	Download
Copyright Notice	Copyright © 2012 Oracle, Inc.
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.glassfish.jersey.media
Name	jersey-media-multipart
Version	2.22
Source Code	Download
Copyright Notice	Copyright © 2012 Oracle, Inc.
Manifest Project URL	http://www.oracle.com/
Manifest License URL	http://glassfish.java.net/public/CDDL+GPL_1_1.html
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	org.jasypt

Name	jasypt
Version	1.9.2
Copyright Notice	Copyright © 2014 The JASYPT team
POM Project URL	http://www.jasypt.org
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt
Embedded License Files	jasypt-1.9.2.jar/META-INF/LICENSE.txt jasypt-1.9.2.jar/META-INF/NOTICE.txt

Group	org.javassist
Name	javassist
Version	3.18.1-GA
Source Code	Download
Copyright Notice	Copyright © 1999 Shigeru Chiba
Manifest License URL	http://www.mozilla.org/MPL/MPL-1.1.html https://www.gnu.org/licenses/lgpl-2.1.html http://www.apache.org/licenses/LICENSE-2.0.txt
POM Project URL	http://www.javassist.org/
POM License	MPL 1.1 – http://www.mozilla.org/MPL/MPL-1.1.html LGPL 2.1 – https://www.gnu.org/licenses/lgpl-2.1.html Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	org.jetbrains.plugins
Name	Groovy
Version	2016.3
Copyright Notice	Copyright © 2000-2010 JetBrains s.r.o.

Group	org.jetbrains.plugins
Name	copyright
Version	2016.3
Copyright Notice	Copyright © 2000-2010 JetBrains s.r.o.

Group	org.jvnet.mimepull
Name	mimepull
Version	1.9.6
Source Code	Download
Copyright Notice	Copyright © 1997-2012 Oracle and/or its affiliates
Manifest Project URL	http://www.oracle.com/

Manifest License URL	https://glassfish.java.net/public/CDDL+GPL_1_1.html
POM Project URL	http://mimepull.java.net
POM License	CDDL 1.1 – http://glassfish.java.net/public/CDDL+GPL_1_1.html

Group	aa
Name	
Version	
Copyright Notice	Copyright © 2019
Manifest Project URL	https://guava-libraries.googlecode.com/
Manifest License URL	http://www.apache.org/licenses/LICENSE-2.0.txt
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Group	org.slf4j
Name	slf4j-api
Version	1.7.21
Copyright Notice	Copyright © 2004-2019 QOS.ch
POM Project URL	http://www.slf4j.org
POM License	MIT – http://www.opensource.org/licenses/mit-license.php

Group	org.slf4j
Name	slf4j-simple
Version	1.7.21
Copyright Notice	Copyright © 2004-2019 QOS.ch
Manifest Project URL	http://www.slf4j.org
Manifest License URL	http://www.opensource.org/licenses/mit-license.php
POM License	MIT – http://www.opensource.org/licenses/mit-license.php

Group	stax
Name	stax-api
Version	1.0.1
Copyright Notice	Copyright © 2004 BEA Systems, Inc.
POM Project URL	https://github.com/codehaus/stax
POM License	Apache License 2.0 – http://www.apache.org/licenses/LICENSE-2.0.txt

Referenced Licenses

- Apache 2.0
- Common Development And Distribution License (CDDL) 1.1
- Common Development And Distribution License (CDDL) 1.0
- Eclipse Public License 1.0
- GNU Lesser General Public License 2.1
- ISC License
- Mozilla Public License 1.1
- MIT License
- Prior Berkeley Software Distribution (BSD) License

Apache 2.0

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent

to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a

Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Common Development And Distribution License (CDDL) 1.1

https://glassfish.java.net/public/CDDL+GPL_1_1.html

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. “Contributor” means each individual or entity that creates or contributes to the creation of Modifications.

1.2. “Contributor Version” means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. “Covered Software” means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. “Executable” means the Covered Software in any form other than Source Code.

1.5. “Initial Developer” means the individual or entity that first makes Original Software available under this License.

1.6. “Larger Work” means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. “License” means this document.

1.8. “Licensable” means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. “Modifications” means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. “Original Software” means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. “Patent Claims” means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. “Source Code” means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. “You” (or “Your”) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, “You” includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered

Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN “AS IS” BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER

EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as “Participant”) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a “commercial item,” as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of “commercial computer software” (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and “commercial computer software documentation” as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

Common Development And Distribution License (CDDL) 1.0

https://glassfish.java.net/public/CDDLv1.0.html

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following: A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications; B. Any new file that contains any part of the Original Software or previous Modification; or C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of:
(1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY. COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.
6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY. UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS. The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS. This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS. As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

Eclipse Public License 1.0

http://www.eclipse.org/legal/epl-v10.html

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

GNU Lesser General Public License 2.1

https://www.gnu.org/licenses/old-licenses/lgpl-2.1.html

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,

INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

ISC License

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Mozilla Public License 1.1

https://www.mozilla.org/en-US/MPL/1.1/

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1 (a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1 (b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2 (a) and 2.2 (b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2 (b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4 (a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Sections 3.1, 3.2, 3.3, 3.4 and 3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. Termination

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declatory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. government end users

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. Miscellaneous

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. Responsibility for claims

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. Multiple-licensed code

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the MPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

MIT License

https://opensource.org/licenses/mit-license.php

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Prior Berkeley Software Distribution (BSD) License

https://en.wikipedia.org/wiki/BSD_licenses

Redistribution and use in source and binary forms are permitted provided that the above copyright notice and this paragraph are duplicated in all such forms and that any documentation, advertising materials, and other materials related to such distribution and use acknowledge that the software was developed by the <organization>. The name of the <organization> may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Git Workflow for Configuration

The purpose of this section is to define an example Git workflow specifically for a Pricef(x) configuration project.

It is strongly recommended to read more about GitFlow before or after this article. It will provide a very good foundational understanding of the typical Git workflow.

This is certainly not the only way to use Git for this purpose; this is merely a recommendation.

Why

There are too many pro's to list here - but I will try to list some.

- Git allows us to track changes made to logics/extensions/parameters etc. This allows us to maintain a history of changes we've made to a project, whereas if we were to work only within the partition there would be no such changelog.
- Another great advantage of this workflow is that it allows team members to do work on the same project in parallel, then use Git to manage merge conflicts and which changes get pushed to dev/qa/prod partitions.
- This workflow will treat the Git repository as the "Single Source of Truth" (SSOT). This means that any change that is made within the partition, but not reflected in the Git repo is deemed to be incorrect and will be lost by the workflow. This means you have one single source, which is transparent and reliable. You do not have to worry about what things are different between partitions.

Recommended Reading

Don't just take my word for it, Git Flow is an industry-accepted standard that has been extensively documented and used already:

- https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow
- https://blog.axosoft.com/gitflow/
- https://datasift.github.io/gitflow/IntroducingGitFlow.html

The post that started it all: https://nvie.com/posts/a-successful-git-branching-model/

In this section

- Git Setup & Usage on Project
- Automated Deployment & Testing
- Git Workflow Scenarios & FAQ
- Helpful Git Commands

Further Interest Reading

There is much debate around the best modern Git workflows. Though this space recommends the standard "Git-flow", there are other more simplified approaches which may better fit some use cases.

- https://githubflow.github.io/
- https://docs.gitlab.com/ee/workflow/gitlab_flow.html
- https://www.freshconsulting.com/git-development-workflows-git-flow-vs-github-flow/

Git Setup & Usage on Project

WORK IN PROGRESS

- How
- Repository Setup
- Further Recommendations
- Tooling

How

The workflow we're using here is very basic and closely follows the standard GitFlow practice.

There is one branch for each of the Pricefx partitions, typically "dev", "qa" and "master" which is the default branch in any new repo and will be used for the production partition.

Each new feature should be contained in its own branch, then merged back into the parent branch.

In every case possible, this work should branch out of "dev" and merge back into "dev". If there is active work to be done on other branches, this is ok but should ideally only be a "hotfix".

I find it much easier to use a Git client for GitFlow, however if you'd like to use CLI, see here for relevant commands.

Below instructions will include CLI commands, though most good Git clients have the GitFlow functionality built in, I would recommend using that instead.

Repository Setup

To start, we need a Git repo. Pricefx currently uses the Atlassian toolset, which has great integration between tools, so we will use BitBucket.

Once the repository is created on Bitbucket, clone it locally and we can begin. Alternatively, create the repo locally.

gitignore for Useless or Sensitive Info

One of the first files to be committed to the repo should be a ".gitignore" file, so that we can tell the repo to ignore sensitive or unnecessary files.

The following .gitignore will stop the repo from tracking the "config.groovy" file which contains authentication information for Pricefx Studio, as well as IntelliJ configuration options. (These should not be shared between team members.)

pom.xml for build configuration:

For help with GitFlow commands, click here.

Initialise GitFlow

This will initialize the repository with default (hence "-d") Git flow setup, and create another branch for qa (master already exists, and develop was created by using init gitflow).

If this is an existing project, and the git workflow is just being implemented, we need to use PFX Studio to fetch all of the logics etc. into the corresponding branches.

Fetch all configuration from Pricefx partition, commit and push to the appropriate branch.

Do this for each existing partition.

New Feature

If developing a new feature or fixing a bug that has an associated issue in Jira, the **new branch name must include the Jira issue ID**. Using GitFlow, this will look like "feature/ProjectCode-IssueID".

This will start a new feature out of the default gitflow develop branch, however other branches can be specified. See above the command reference.

By using the Jira issue ID as the branch name, we allow quick navigation to the branch in Bitbucket web UI straight out of the Jira issue via a hyperlink.

Hotfix (might not have associated Jira issue)

If developing a quick fix / feature that does not have an associated issue in Jira, we will use the default GitFlow branch name "hotfix/some_meaningful_name".

When committing changes to the repo, **always include the Jira Issue ID in the commit message** where possible. This will also provide hyperlinks straight from the Jira issue card to that commit in the Bitbucket web UI.

Commit changes as many times as needed during development.

Upon completion, we need to "finish" the feature/hotfix.

Finish Feature/Hotfix

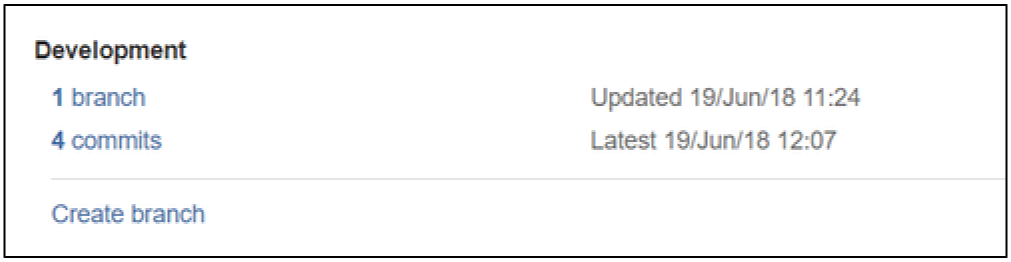
After the changes are committed and the branch is no longer needed, use GitFlow to "finish" the feature/hotfix. This will ask what branch you would like to merge back into (default is dev) and will merge changes into the dev branch and delete the feature branch.

> Only merge to partition branches if changes are deployed and stable. (Even if not complete, feature branches should never merge to dev/qa branches with partial commits.)

It is our own responsibility to ensure that any change pushed to the dev branch are also deployed to the dev partition in Pricefx.

If you're willing to get hands-on with CI/CD tooling and you have the authority of the project architect – see Continuous Delivery in the Tooling section to see how to do **automagic deployment**.

If you put the Jira Issue ID in the branch name, and in each of your commits, your Jira card should look something like the below. This is incredibly useful for review by yourself and peers.



Development

1 branch — Updated 19/Jun/18 11:24

4 commits — Latest 19/Jun/18 12:07

Create branch

This new feature/hotfix branch can be used for as long as it takes to complete the task.

This will keep all work for that feature out of the active partition-representative branches.

It will look something like this in a graphical client:

Code changes stay in branches and commits relevant to that issue, and out of the way of other issues and team members' work. Beautiful right!?

Further Recommendations

Don't be that guy!

Never commit straight to a dev/qa/master branch.

That would make this whole workflow pointless, and makes managing merge conflicts between team members really hard.

Always start a new feature or hotfix branch.

Tooling

The following is a list of useful tools involved in making this happen.

Tool Type	Resource
Git GUI Client	https://www.gitkraken.com/ https://www.sourcetreeapp.com/ https://www.git-tower.com/
Development Environment	https://www.jetbrains.com/idea/download/
Pricefx Development Extensions	Pricefx Studio
Continuous Integration / Delivery	Continuous Delivery for Client Configuration Project https://jenkins.pricefx.eu/jenkins/ TDD4C

Automated Deployment & Testing

WORK IN PROGRESS

Using pfx-package to automatically import/export configurations to a partition.

Note: the pfx-package tool does not cover the entirety of a pricefx partition config. It does do most things though, and certainly makes this process much easier than doing manually.

Resources

PFXPACKAGE (PFAUT) PROJECT

Project: https://pricefx.atlassian.net/projects/PFAUT
Repo: https://bitbucket.org/pricefx/pfxpackage
Build: https://jenkins.pricefx.eu/jenkins/job/pfxpackage/

EXAMPLE JENKINS DEPLOYMENT PROJECTS:

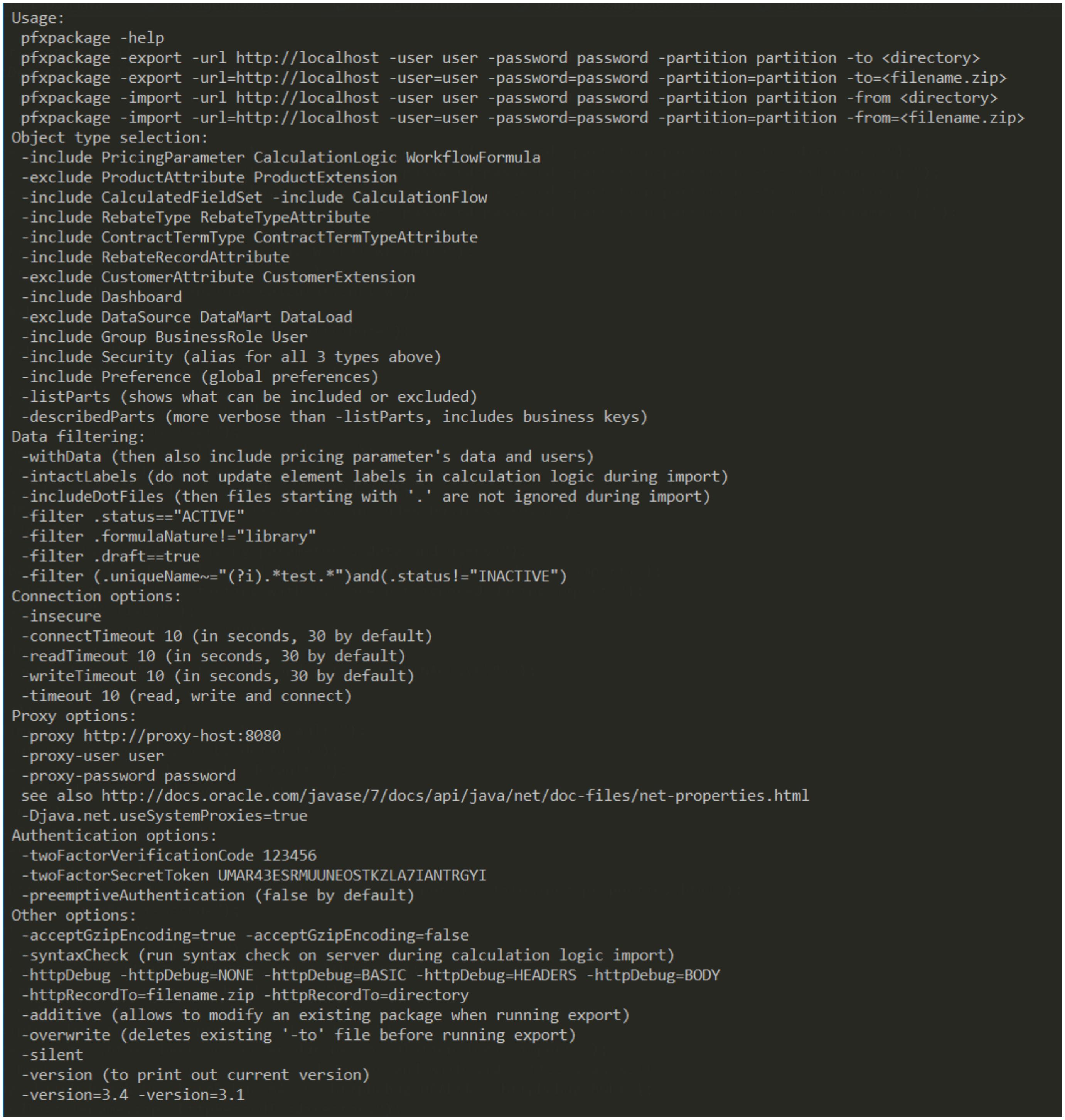
https://jenkins.pricefx.eu/jenkins/view/Config-CD/

Any job with "configuration-deploy" string in name will be appear in above group.

USEFUL COMMANDS:

PfxPackage:

Usage Help



```
Usage:
 pfxpackage -help
 pfxpackage -export -url http://localhost -user user -password password -partition partition -to <directory>
 pfxpackage -export -url=http://localhost -user=user -password=password -partition=partition -to=<filename.zip>
 pfxpackage -import -url http://localhost -user user -password password -partition partition -from <directory>
 pfxpackage -import -url=http://localhost -user=user -password=password -partition=partition -from=<filename.zip>
Object type selection:
 -include PricingParameter CalculationLogic WorkflowFormula
 -exclude ProductAttribute ProductExtension
 -include CalculatedFieldSet -include CalculationFlow
 -include RebateType RebateTypeAttribute
 -include ContractTermType ContractTermTypeAttribute
 -include RebateRecordAttribute
 -exclude CustomerAttribute CustomerExtension
 -include Dashboard
 -exclude DataSource DataMart DataLoad
 -include Group BusinessRole User
 -include Security (alias for all 3 types above)
 -include Preference (global preferences)
 -listParts (shows what can be included or excluded)
 -describedParts (more verbose than -listParts, includes business keys)
Data filtering:
 -withData (then also include pricing parameter's data and users)
 -intactLabels (do not update element labels in calculation logic during import)
 -includeDotFiles (then files starting with '.' are not ignored during import)
 -filter .status=="ACTIVE"
 -filter .formulaNature!="library"
 -filter .draft==true
 -filter (.uniqueName~="(?i).*test.*")and(.status!="INACTIVE")
Connection options:
 -insecure
 -connectTimeout 10 (in seconds, 30 by default)
 -readTimeout 10 (in seconds, 30 by default)
 -writeTimeout 10 (in seconds, 30 by default)
 -timeout 10 (read, write and connect)
Proxy options:
 -proxy http://proxy-host:8080
 -proxy-user user
 -proxy-password password
 see also http://docs.oracle.com/javase/7/docs/api/java/net/doc-files/net-properties.html
 -Djava.net.useSystemProxies=true
Authentication options:
 -twoFactorVerificationCode 123456
 -twoFactorSecretToken UMAR43ESRMUUNEOSTKZLA7IANTRGYI
 -preemptiveAuthentication (false by default)
Other options:
 -acceptGzipEncoding=true -acceptGzipEncoding=false
 -syntaxCheck (run syntax check on server during calculation logic import)
 -httpDebug -httpDebug=NONE -httpDebug=BASIC -httpDebug=HEADERS -httpDebug=BODY
 -httpRecordTo=filename.zip -httpRecordTo=directory
 -additive (allows to modify an existing package when running export)
 -overwrite (deletes existing '-to' file before running export)
 -silent
 -version (to print out current version)
 -version=3.4 -version=3.1
```

Git Workflow Scenarios & FAQ

A place to document those tricky "how do I handle this case?" workflow scenarios for Pricefx configuration Git workflow.

Scenario Demonstrations

Project Setup

- Init master
- Gitignore file
- Set global autoclrf
- Create branch for develop
- Init GitFlow in client

Feature

- Create new branch
- Stage good changes
- Commit staged changes
- Commit more changes to same feature
- PR / merge feature to develop

Feature in Parallel with Teammate

Hotfix

- Branch from topmost environment in which bug exists
- Commit to hotfix branch
- Merge hotfix to env and back down the tree

Release

- New release branch from develop
- Bugfix on release
- Merge to develop
- Merge to master upon completion
- Tag master with release number

Pull Request With Conflicts

- New feature branch from develop
- Teammate merges other feature to develop
- Commit own changes in feature
- Submit PR / Merge for own feature – see conflicts
- Merge develop to feature
- Resolve conflicts in feature branch
- Commit resolved conflicts and update PR

Squash Commit Series & Force Push

- New feature branch
- Series of commits to branch
- Push commits
- Squash commits
- Force push

Force Move of Branch Ref – With Backup Ref to Old

- "Accidental" commit direct to master
- New branch at new commit
- Reset master to correct location
- Force push master
- Check out corrected branch – carry on

FAQ

When should a feature branch be closed?

A feature branch should close when it is merged to a major branch. It is fine to re-open a feature branch if necessary, or even to create a new one with suffix -2 etc.

Should we close release branches?

Release branches should be closed when they're merged to master. Version release date is tracked using Git tags. Tag the merge-commit on master with the version number.

My Pull Request Has Merge Conflicts, What Should I Do?

Merge the destination branch into the source (feature) branch first. Resolve the merge conflicts in the source branch and re-commit. The PR will automatically see the updated commit with the conflicts resolved.

IDE Usage and Hints

Git Annotate
Git Show History
Git Compare With

Helpful Git Commands

Configuration Packaging

The **pfxpackage** is a tool that exports and imports selected customization/configuration parts to/from a Package Structure. The Package is an organized and partition-neutral representation of PFX objects. The package format is canonical: the same "user-observed" configuration always yields to the same binary package.

Using this tool you can:

- Export packages from several related PFX instances/partitions (such as customer environments) and compare them using GUI tools (e.g. Meld, WinDiff, Total Commander, IDEA).
- Save any package to a Git repository to represent a particular state.
- Import the package to any partition.
  - This way you can apply the same customization/configuration to multiple PFX partitions (environments).
- Delete package data from any partition.

The package structure and data are independent of a specific PFX partition/instance and they are PFX API version tolerant.

- Quick Start
- Package Structure

> To read the internal technical and implementation notes, see the Engineering space.

Quick Start

1. Make sure that JRE 1.8 or newer is installed.
2. Download and unpack the distribution:
   - On Windows: download https://nexus.pricefx.net/tools/net/pricefx/pricefx-pckg/3.5/pricefx-pckg-3.5.zip and unzip it to a directory of your choice.
   - On Mac/Linux: download https://nexus.pricefx.net/tools/net/pricefx/pricefx-pckg/3.5/pricefx-pckg-3.5.tar.gz and run:
     tar xvf <downloaded file> -C "Your Choice"

The tool executable is called **pfxpackage**, it is in "Your Choice"/pfxpackage*/bin/pfxpackage.

To get help, run `pfxpackage`.

Examples:

```
pfxpackage -export -url http://localhost -user sranka -password sranka
-partition sranka -to pfcd3452_logics.zip -overwrite -include
CalculationLogic -filter
'(.uniqueName=="pfcd3452")or(.uniqueName=="pfcd3452_matrix")'

pfxpackage -export -url http://localhost -user sranka -password sranka
-partition sranka -to outDirectory -filter '.uniqueName=="Pepa2.2"'
-include PricingParameter

pfxpackage -import -url http://localhost -user sranka -password sranka
-partition sranka2 -from myTestPackage.zip -exclude CalculationLogic

pfxpackage -listParts

pfxpackage -delete -url http://localhost -user sranka -password sranka
-partition sranka2 -y -from myTestPackage.zip -exclude CalculationLogic

pfxpackage -delete -url http://localhost -user sranka -password sranka
-partition sranka2 -y -type CalculationLogic -uniqueName=MyLogic
-validFrom=2019-02-18
```

Package Structure

The configuration package structure is human readable and self-describing.

Design Principles

Package structure:

- Singular and camel case are used.
- Files have extensions, directories do not.
- The first level directories represent particular configuration type, the directory name is self-describing.
    - Data and metadata are stored in these directories; data are always optional.
    - For simple types, the directory contains only files with metadata.
    - For more complex structures, nested directories are created.
    - Nested files/directories are named after the entity business key whenever possible.

Exported files:

- JSON files are created:
    - Pretty printed with indentation of 2 spaces.
    - With canonical serialization (the JSON tree is fully sorted in alphabetical order).
    - UNIX LF
- CSV files when created:
    - Always have a header line.
    - Sorted by the business key (depends on the exported data).
    - UNIX LF, comma-separated fields.
- References to other entities used in files are expressed with a business key, database ID is never used.

Structure

Names are constant or use brackets from Utility Conventions: square brackets [optional], angle brackets <required>, curly brackets {default values}, parenthesis (miscellaneous info).

```
AdvancedConfiguration
  <key>.txt
CalculatedFieldSet
  <label>[+<targetDate.format("yyyy-MM-dd")].json
CalculationFlow
  <uniqueName>+<Draft|Deployed>.json
CalculationLogic
  <uniqueName>+<validAfter.format("yyyy-MM-dd")>[+<simulationSet>]
    elements (directory)
      elem1.groovy (.groovy files for groovy FUNCTION expressions)
      elem2.formula (formula expression)
    logic.json (entity JSON with formula expression)
ContractTermType
  [<uniqueName>.json]
ContractTermTypeAttribute
  [attribute1-30.json]
  [formulaName.json]
  [label.json]
  [waterfallElement.json]
CustomerAttribute
  [attribute1-30.json]
  [customerId.json]
  [isParent.json]
  [name.json]
  [parentId.json]
CustomerExtension
  <label>
    attributes (directory)
      [<attributeName>.json]
    extension.json
Dashboard
  <uniqueName>.json
DataLoad
  <target>+<type>+<label>
    dataload.json
    schedules.csv
DataMart
  <uniqueName>
    fields
      <customAttribute>.json
      <systemAuxilaryAttribute>.json
    datamart.json
    ranks.csv
DataSource
  <uniqueName>
    fields
      <customAttribute>.json
      <systemAuxilaryAttribute>.json
    source.json
    ranks.csv
```

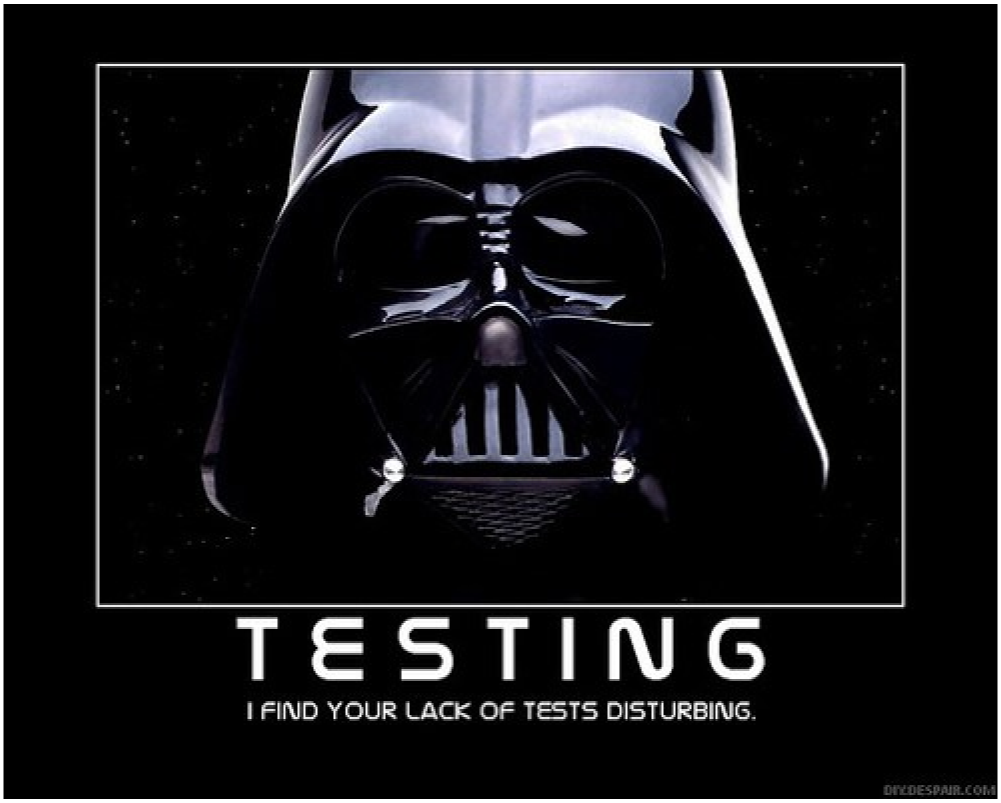
```
MessageTemplate
  [languagesForMsgTemplates.json]
  [<templateKey>_<locale>.html]
  [<templateKey>_<locale>.txt]
Preference
  <preferenceType>
    <preferenceName>.json
PriceRecordAttribute
  [<attributeName>.json]
PricingParameter
  <uniqueName>+<validAfter.format("yyyy-MM-dd")>[+<simulationSet>]
    parameter.json (entity JSON)
    matrix (directory of matrix column definitions)
      <field>.json (matrix field definitions)
    data_simple.csv (data of simple type, requires -withData option)
    data_range.csv (data of range type, requires -withData option)
    data_matrix.csv (data of matrix type with 1 key, requires -withData
option)
    data_matrix<2|3|4|5|6>key.csv (data of matrix type with 2-6 keys,
requires -withData option)
  tree.csv (structure of the tree)
ProductAttribute
  [attribute1-30.json]
  [currency.json]
  [label.json]
  [sku.json]
  [unitOfMeasure.json]
ProductExtension
  <label>
    attributes (directory)
      [<attributeName>.json]
    extension.json
RebateAgreementTemplate
  [<uniqueName>.json]
RebateAgreementTemplateAttribute
  [attribute1-30.json]
  [formulaName.json]
  [label.json]
  [lastUpdateDate.json]
  [uniqueName.json]
RebateRecordAttribute
  <RebateRecordSetLabel> (usually Default)
    [<attributeName>.json]
RebateType
  [<uniqueName>.json]
RebateTypeAttribute
  [attribute1-30.json]
  [formulaName.json]
  [label.json]
  [waterfallElement.json]
```

```
SavedChart
  [chartDefinition.json]
Security
  BusinessRole
    [<uniqueName>.json]
  User
    [<loginName>.json]
  users.csv (use to import passwords, CSV header line is then
"loginName,password", more columns for user fields can be added as well)
  userGroups.csv
WorkflowFormula
  <uniqueName>+<validAfter.format("yyyy-MM-dd")>
    elements (directory)
      <elemementName>.groovy (.groovy files for groovy FUNCTION
expressions)
```

```
    logic.json (entity JSON with formula expression)
.info (information about export version and possibly more)
```

TDD4C

TDD4C stands for Test Driven Development 4 Consulting. It is a tiny project that allows logics (formulas) to be run locally in your favourite Java IDE or on any Continuous Integration instance (e.g. Jenkins, Team City, ...). The main goal is to provide simple and easy way of writing unit tests for existing and newly created logics.

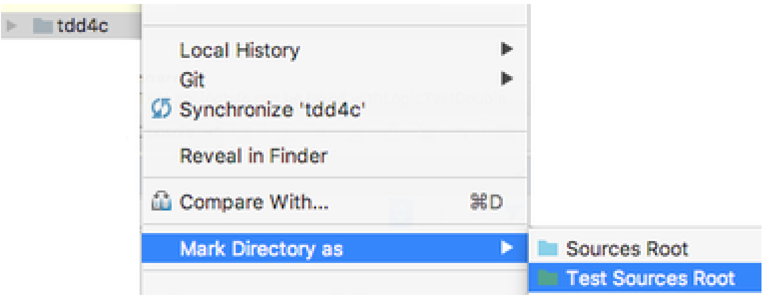


Setting up a New Project

In order to introduce TDD4C to you logic, you need to use Pricefx Studio or packaging to be able to fetch logics from the server. Then, set up a Git repo for your project, which will allow you to have continuous integration job on PFX Jenkins at https://nexus.pricefx.net/jenkins/view/tdd4c/.

Create a Root pom.xml File

1. Create a Maven metadata file where you enable Groovy usage and pull all the dependencies. In the root folder of your project, create the pom.xml file like this one (substitute "myproject" with you project name).
2. Create the tdd4c folder in the project's root directory.
3. Mark it as the test sources root in you favourite IDE.

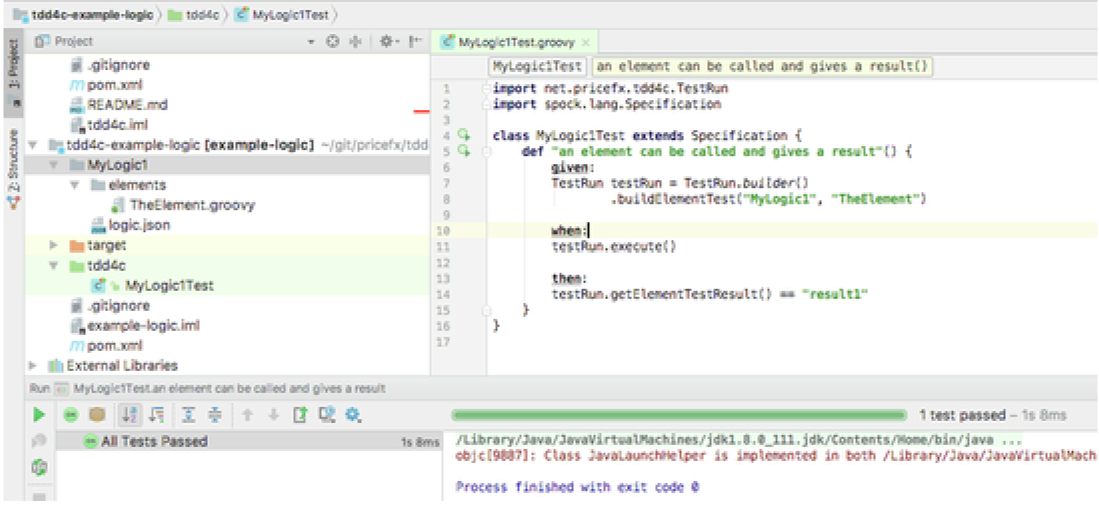


Running My First Test

For writing tests, we prefer Groovy and JUnit. Although Groovy and JUnit are sufficient, we also use the Spock framework as it brings a huge amount of very helpful features to the tests.

To get started:

1. Clone the example project from: https://bitbucket.org/pricefx/tdd4c-example-logic
2. Set up your IDE.
3. Open tdd4c/MyFirst1Test.groovy
4. Run it.



```
import net.pricefx.tdd4c.TestRun
import spock.lang.Specification

class MyLogic1Test extends Specification {
    def "an element can be called and gives a result"() {
        given:
        TestRun testRun = TestRun.builder()
                .buildElementTest("MyLogic1", "TheElement")

        when:
        testRun.execute()

        then:
        testRun.getElementTestResult() == "result1"
    }
}
```

```
/Library/Java/JavaVirtualMachines/jdk1.8.0_111.jdk/Contents/Home/bin/java ...
objc[9887]: Class JavaLaunchHelper is implemented in both /Library/Java/JavaVirtualMach

Process finished with exit code 0
```

The test itself instantiates an instance of the TestRun class which is the tiny (test run) context holder that evaluates all the formulas and stores element results.

When executed, you can query the element results.

> Explore the other tests in the repository.

Setting Element Results

TDD4C runs only Groovy elements. All other elements must be set in advance, otherwise the logic execution will fail.

To do so, use the .withElementResult method on TestRun. Setting the result for the Groovy element will ensure that the Groovy element will not be evaluated and the provided value will be used instead.

Creating Test Doubles

TestDoubles are objects that we pass to the test run that interact with the code under test, i.e. all injected variables (api, quoteProcessor, rebateManager, ...).

The test double can be a fake or a mock. By default, when no test doubles are provided, some default fakes are injected to the elements to allow the code execution. However, the test doubles need to be configured per test.

I Warned You!

Whenever a test double is needed to provide some dependency, it is strongly recommended that you change the code so that it passes the dependency value to a new function as a parameter and then you test just this new function. This way it is much easier to test and maintain in future.

Map Based Fakes

By default, we use fake objects created as maps. These default fake objects are defined in the upstream tdd4c library. In a test, you can override properties on these fakes to provide the test specific behavior.

This code will ensure that the default fake for API variable has overridden the find function that will be called in a test. Whenever MyLogic3 is executed, it takes the default map based fake and adds the behavior specified in .withLogicTestDouble. This way you can create test doubles with minimal effort.

There is also the .withElementTestDouble method. It specifies the overrides for the default + logic test double to be provided in a single element.

The methods api.getElement and api.getBinding are handled gracefully in TestRun and should always provide the expected result. There should not be any need to implement it manually.

Object Based Fakes

If map based fakes cannot be used, you can provide object based implementation. This should be needed only when there is a collision of some Groovy default method (e.g. groupBy()) with the defined test double. No merging of the element test double with the logic test doubles will happen with the object based fakes.

Mocks

There are a few PFX interfaces bundled with the tdd4c library. PublishedApi, DatamartContext and TableContext to name some of them. You can use the Mockito library to create mocks and pass them as the test double. No merging of the element test double with the logic test double will happen in this case.

Testing Functions

Whenever you need to test a function defined in an element, you can do it this way:

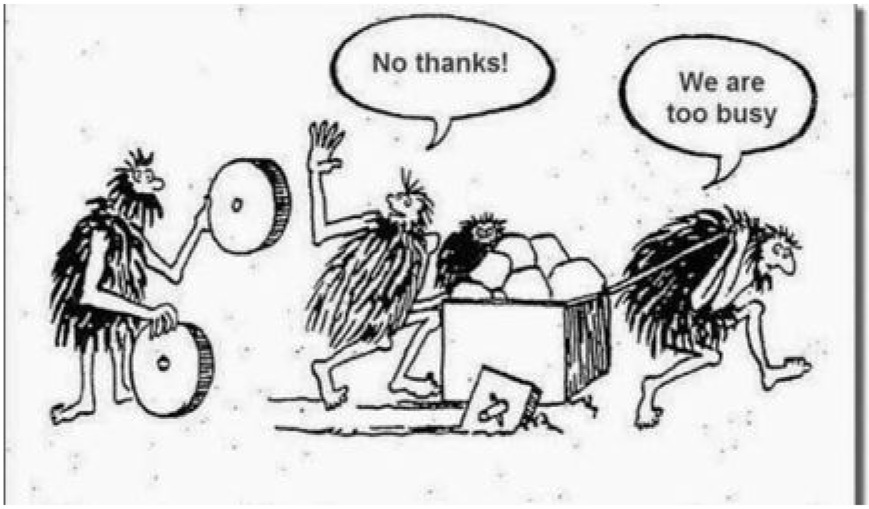
FAQ

I would like to do X rather than this!

No please. https://8thlight.com/blog/uncle-bob/2012/01/11/Flipping-the-Bit.html

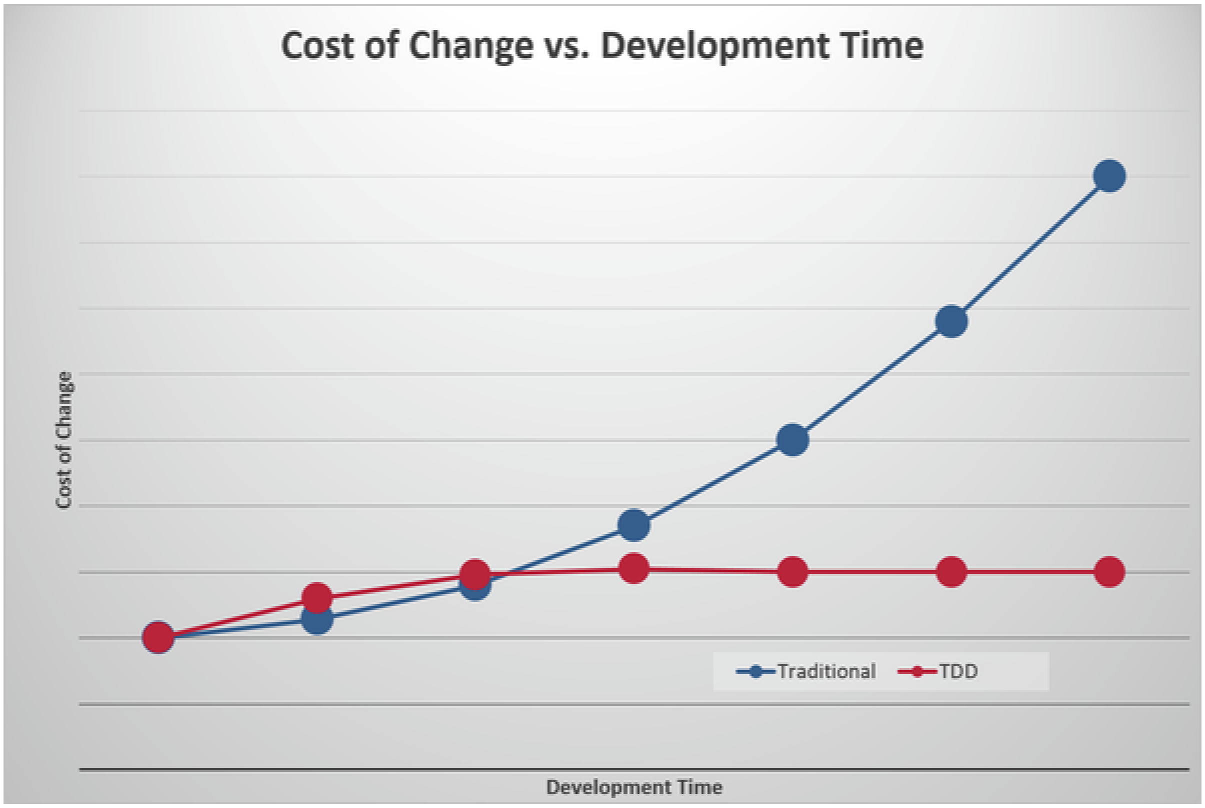
We do not have time do do this!

No one has.



Really?

Yes.



How to explain that this might be useful?

Feel free to use these slides:

TDD4C_final.pdf

Contributing to This Project

TDD4C lives in https://bitbucket.org/pricefx/tdd4c and code contributions are warmly welcome 🙂

Clone the repo, create a pull request and add Tomas Lamr as reviewer.

PFX Changelogs

To get the latest hot-patch changelog (i.e. changes done after the first release of the major version) for the Bramble 3.7 version, go to:

- https://nexus.pricefx.net/pricefx-master-changelog.html

Older, static versions are stored here:

- El Presidente 3.6 Changelog

- Salty Dog 3.5 Changelog
- Cuba Libre 3.4 Changelog
- Cosmopolitan 3.3 Changelog
- Mojito 3.2 SE Changelog
- Mojito 3.2 Changelog
- Mai Tai 3.1 pricefx-server Changelog
- Mai Tai 3.1 pricefx-web Changelog
- SeaBreeze 3.0 pricefx-server Changelog
- SeaBreeze 3.0 pricefx-web Changelog

To understand the changelog better, read this topic at the Q&A Forum: How do I tell which fixes from the change log made it into a specific build?

El Presidente 3.6 Changelog

Commit	Date	Issue	Message
155ec838d57059f	2019-03-08 15:35:40	PFCD-4164	PFCD-4164: reverting closing the TableContext (H2 DB) on formula engine reset
7d70c0e1a0df608	2019-03-05 12:57:51	PFCD-4722	PFCD-4722: Bug fix after revert
892c4c1a5568e86	2019-03-05 12:50:32	PFCD-4722	Revert "PFCD-4722: Default chart currency" This reverts commit 487315b
0981fb86aed995e	2019-03-05 12:45:44	PFCD-4722	PFCD-4722: default chart currency is coming from source.baseCurrency or partition.baseCurrency
487315bfee4365f	2019-02-28 08:44:50	PFCD-4722	PFCD-4722: Default chart currency
4cb383dddb3291e	2019-03-12 09:24:12	PFCD-4863	PFCD-4863: Added admin access for systmadmin to see QTTs
2b505146d5b619a	2019-03-12 15:56:19	PFCD-4906	PFCD-5286 fix regression from PFCD-4906
9f60515fdfd8136	2019-04-10 17:13:21	PFCD-4926	PFCD-4926 Remove unused import
3f3fca87773657e	2019-04-10 17:12:31	PFCD-4926	PFCD-4926 Improve calculation DL cleanup stability on error/cancellation and also try to be more proactive to catch interrupts outside of the io sections
bdd2af182c114f4	2019-04-10 17:07:51	PFCD-4926	PFCD-4926 Improve status accuracy and logging for DataLoad cancellations Also moved an interrupt swallowing statement from one "catch" clause to the "final" clause shortly before the most interrupt sensitive calls (save and flush)
56f5a2a99781aa7	2019-04-10 17:03:33	PFCD-4926	PFCD-4926 Try to set the correct status for DataLoad cancellations
7ba9f9523f585ad	2019-04-10 17:02:24	PFCD-4926	PFCD-4926 Avoid IllegalStateException in FileBackedMVStore.rollback by checking that the db/store isn't closed
2343a249b7720b4	2019-04-01 14:38:28	PFCD-4926	PFCD-4926 Int test for datamart.canceldataload
6dfedfbf40a5b25	2019-04-01 14:37:52	PFCD-4926	PFCD-4926 Fix inconsistent DataLoad status on cancel request When handling a "datamart.canceldataload" request, in case the DL is in a non-final state but no job is actually pending or processing, we set the DL to CANCELLED.
4c1bb6732eca683	2019-04-01 07:38:52	PFCD-4926	PFCD-4926 IntTest tweaks while working on CancelDataLoad IT
5952a4acce3d13e	2019-03-19 12:35:02	PFCD-4929	PFCD-4929 improved UI label
c611a924f609c84	2019-03-05 15:59:37	PFCD-4929	PFCD-4929 Make DataTable's initialDataGrouping option fully functional This option causes the Data Table to be initially displayed with all group nodes collapsed. This option replaces the grouping prefs that were saved so far. The appropriate Groovy ChartBuilder hint is generated so the grouping is also applied in dashboards.
cd2dace012adc46	2019-03-18 12:45:42	PFCD-4965	PFCD-4965 prevent scrollbar blinking
6215d09bb5e6ddb	2019-02-20 09:03:43	PFCD-4971	PFCD-4971 Dim Filters not visible without refreshing

43951ad7a2f16ad	2019-02-19 09:31:05	PFCD-4971	PFCD-4971 Dim Filters not visible without refreshing
017a04fbca5c537	2019-03-19 11:04:18	PFCD-5034	PFCD-5034 Filter Highcharts SVG content using OWASP Sanitizer
7463e5aa0c85908	2019-03-19 10:34:00	PFCD-5034	PFCD-5034 code tidying
d42d7138bdd8250	2019-02-14 15:38:36	PFCD-5061	PFCD-5061: RR roles fixed
180a070a0de3cde	2019-02-27 08:51:27	PFCD-5064	PFCD-5064: Improved UI in case of no PA access for PDE and Dashboards. My Chart (start page) and DATA_MANAGER_PAGE access fixed.
4b51ece9aad16af	2019-03-07 13:53:30	PFCD-5086	PFCD-5086: Previous merging fixed
7fd94f89fc227e2	2019-03-06 18:03:28	PFCD-5098	PFCD-5098 Second take at fixing dashboard Excel export of Long values
462018adac30038	2019-03-06 17:24:05	PFCD-5098	PFCD-5252: reverting back part of PFCD-5098
c60da6b4bb1332b	2019-02-22 14:54:26	PFCD-5105	PFCD-5105 Rebates Not Displaying Units for Multi-Tier Entry
010404184f1de13	2019-03-08 12:15:41	PFCD-5113	PFCD-5113 Do not show percent values as decimal numbers
5dcd43e2588cb60	2019-02-28 15:10:18	PFCD-5129	PFCD-5207 Partial fix for Waterfall data export to Excel The exported columns may not be ordered as they are in the UI, but at least it will not break the query. A full fix will be merged into the next release. See PFCD-5129.
66d967fb2f5b194	2019-03-11 08:02:40	PFCD-5150	PFCD-5150 Filters - Customers, Products: When typing char in "Last Update", application falls with Error
55cdd754b5f886e	2019-02-25 13:02:20	PFCD-5150	PFCD-5150 typo PFCD-5150 typo
7d1174082d2fae0	2019-02-25 12:57:17	PFCD-5150	PFCD-5150 Validates date filter before it's send to server
fc4ecebf628245d	2019-03-19 07:35:25	PFCD-5157	PFCD-5157: Merging selected Prefereces: improvements based on PR comments
684805a58a38539	2019-03-14 13:52:40	PFCD-5157	PFCD-5157: Merging selected Prefereces and savedGridState for Q/CT/RA - unless you manually change columns.
d113623543a46c9	2019-03-07 15:21:04	PFCD-5172	PFCD-5172: NPE fix - only applies to STN
d9db5c9d611cfcf	2019-02-26 13:30:11	PFCD-5178	PFCD-5178 Create filtered Pricelist from LPG
261b4a56f30d91d	2019-03-12 07:11:46	PFCD-5185	PFCD-5185: Exposing base UOM and currency in DM editor + IT
7278465e538e576	2019-02-27 13:53:53	PFCD-5195	PFCD-5195: MassUpdate readonly access for support role
5dcd43e2588cb60	2019-02-28 15:10:18	PFCD-5207	PFCD-5207 Partial fix for Waterfall data export to Excel The exported columns may not be ordered as they are in the UI, but at least it will not break the query. A full fix will be merged into the next release. See PFCD-5129.
884b057dc6d7aff	2019-03-08 08:45:55	PFCD-5215	PFCD-5215 fix
bcc3c90631fd3d0	2019-03-04 09:31:07	PFCD-5215	PFCD-5215 disable filter on typedId, fix sorting on typedId
2639162600462e2	2019-03-05 08:09:29	PFCD-5218	PFCD-5218 Datetime in Matrix is shown only as Date
93ce9e1872d1028	2019-03-06 11:40:37	PFCD-5219	PFCD-5219 Unable to reset column customization of Rebate Attribute Template
00630084090523e	2019-03-27 15:09:51	PFCD-5221	PFCD-5221: fix NPE in previous commit
9e4e8c4b157ebdb	2019-03-14 23:58:49	PFCD-5221	PFCD-5221: Hide CalcStep 'user entry' in UI

7f2df0346673fae	2019-03-14 23:51:17	PFCD-5221	PFCD-5221: showAlertMessage()
cc9483263e17694	2019-03-14 19:25:21	PFCD-5221	PFCD-5221: Hide CalcStep 'user entry' in UI
3a3ba7224a8e529	2019-03-09 21:44:22	PFCD-5221	PFCD-5221: PO/R tweaks
833e962b85cf23e	2019-03-03 23:12:01	PFCD-5221	PFCD-5221-advanced-maths-lib: insitiate Matrix2D from list of maps (from groovy)
c607c720014f4fc	2019-03-06 13:09:27	PFCD-5222	PFCD-5222 Configuration: Missing label / translation
472cacb05b31b44	2019-03-04 10:02:21	PFCD-5224	PFCD-5224 Q/CT/RBA detail backend call disabling panel instead of showing dialog
0709f0159db37e5	2019-03-07 14:55:16	PFCD-5245	PFCD-5245: IT fix for Oracle
02a28931dfa1ff7	2019-03-07 12:27:16	PFCD-5245	PFCD-5245: Fix when no default quotetype for getWorkflowFormula
c4a553d0d2bac86	2019-03-07 10:09:42	PFCD-5245	PFCD-5245: Quote Processing more errorproof
b2e5810888d5aad	2019-03-05 08:11:12	PFCD-5245	PFCD-5245: Default quote type is created during migration or when new partition is being created
b36d9b5abf39ce6	2019-03-06 13:17:18	PFCD-5247	PFCD-5247 Advanced filter doesn't work properly for Match None
f0bc1903c86b885	2019-03-06 10:49:57	PFCD-5247	PFCD-5247 Advanced filter doesn't work properly for Match None
304145c8147f925	2019-03-06 12:55:25	PFCD-5249	PFCD-5249 wrong default attributeExtension label
ca8dc97046ad901	2019-03-06 13:26:37	PFCD-5251	PFCD-5251 clear api key cache
462018adac30038	2019-03-06 17:24:05	PFCD-5252	PFCD-5252: reverting back part of PFCD-5098
abc95608dfe8c0f	2019-03-06 08:47:42	PFCD-5253	PFCD-5253 avoid invalid zipbomb detection
53bc37cd925f35e	2019-03-06 10:49:27	PFCD-5255	Revert: PFCD-5255 wrong images path in result matrices (378ace41)
311a7e4b401032d	2019-03-06 10:48:33	PFCD-5255	PFCD-5255 handle img paths in web
378ace4135689d1	2019-03-06 09:40:57	PFCD-5255	PFCD-5255 wrong images path in result matrices
349f0b0a7845090	2019-03-07 11:45:05	PFCD-5257	PFCD-5257 broken custom date handling in wf window
7278f45c61ba796	2019-03-07 09:07:07	PFCD-5260	PFCD-5260 Holes in columns are skipped when parsing excel file through excel file input
534565f0ce671e8	2019-03-08 11:36:30	PFCD-5261	PFCD-5261 oops
95b268f9c73c603	2019-03-08 11:34:38	PFCD-5261	PFCD-5261 fix transient filter state handling
890f5c242643b0d	2019-03-07 13:37:47	PFCD-5262	PFCD-5262 Submit button on Contract is not disabled when CRITICAL alerts exists
a3151ad4c7c0b6d	2019-03-19 09:50:34	PFCD-5263	PFCD-5263 remove confusing setting for email processor
e90af7bbf8cfbb9	2019-03-08 12:57:02	PFCD-5263	PFCD-5263 make integrations great again
b094f3a7370acf5	2019-03-08 12:07:35	PFCD-5263	PFCD-5263 disabled help icon missing
82d386ef646962a	2019-03-08 11:58:46	PFCD-5263	PFCD-5263 Better and easier setup of event processing

4e29d8eb0e4c132	2019-03-19 12:19:48	PFCD-5265	PFCD-5265 small bit missing in previous commit Before committing I had restored the 'else' clause one last time to confirm that the int-tests would fail... then pushed that version by mistake.
a931fdc4fddb4ab	2019-03-14 15:16:03	PFCD-5265	PFCD-5265 fix DM refresh and querying of DataTime fields from 2ry DSes We were missing some dependencies in the field definitions and when building the dependency graph
f7d05a43521ed1a	2019-03-07 17:13:21	PFCD-5265	PFCD-5265 tentative int test change for a DM refresh issue
ccfcb261917cc6f	2019-03-27 13:06:34	PFCD-5267	PFCD-5267 Improve dead node detection
d0688cbca0b9ec8	2019-03-22 12:58:32	PFCD-5267	PFCD-5267 - improve thread on dead node detection. Do only when also lastUpdateDate is unchanged for hours
1e478e3402de8e0	2019-03-15 14:40:27	PFCD-5269	PFCD-5269 PB > Convert LPG to PL: Filters should not be converted.
a7bad95a7cfd1c0	2019-03-11 14:06:30	PFCD-5277	PFCD-5277 - The new typedId column is downloaded to Excel even if not visible Do not export typedId column if it is hidden.
d713a198f6cca88	2019-03-12 16:27:56	PFCD-5286	PFCD-5287 Disable PFCD-5286 by default
8784e4fb087f9f1	2019-03-12 16:05:28	PFCD-5286	PFCD-5286 distinguish integration user
2b505146d5b619a	2019-03-12 15:56:19	PFCD-5286	PFCD-5286 fix regression from PFCD-4906
18446ebacbe459b	2019-03-13 15:16:29	PFCD-5287	PFCD-5287 allow update only for data interation role
0340b5235b27351	2019-03-13 05:24:54	PFCD-5287	PFCD-5287 better logging
d713a198f6cca88	2019-03-12 16:27:56	PFCD-5287	PFCD-5287 Disable PFCD-5286 by default
a76d49bd5846dac	2019-03-25 09:53:07	PFCD-5288	PFCD-5288 Price API: By default do not store full Price API record in PCOMP table
5c4364e05fb9b8e	2019-04-08 11:41:20	PFCD-5290	PFCD-5290 - [WEB ONLY] - Customer/Products master table - values restricted by list - visibility
72d27ae51f7b7f2	2019-03-19 15:02:22	PFCD-5297	PFCD-5297 PB: Converting fails when you convert simple LPG to PL from LPG detail view
016e0d17182bdb9	2019-03-15 09:22:29	PFCD-5297	PFCD-5297 PB: Converting fails when you convert simple LPG to PL from LPG detail view
e1ec465ae509be7	2019-03-22 13:37:13	PFCD-5300	PFCD-5300 Excel Client crashing when processing proxy script on Excel 64-bit PFCD-5324 EC crashing when displaying list of PL entries for selection EC Commit: 704807bafffe33e6e7f520fd9167e9c2b289aa32 [704807b]
832590d615cac97	2019-03-26 13:08:36	PFCD-5301	PFCD-5301 PA-create-indexes-on-ds-fks
fdafcfa7cc4fb03	2019-03-19 16:44:46	PFCD-5312	PFCD-5312 Fix cache invalidation for Rollups and other queries
5fbd22cc9f4790a	2019-03-20 13:39:46	PFCD-5315	PFCD-5315 perf
696e52bcb57eddf	2019-03-20 09:40:57	PFCD-5318	PFCD-5318 Exception thrown from Price list workflow is not visible to user
e1ec465ae509be7	2019-03-22 13:37:13	PFCD-5324	PFCD-5300 Excel Client crashing when processing proxy script on Excel 64-bit PFCD-5324 EC crashing when displaying list of PL entries for selection EC Commit: 704807bafffe33e6e7f520fd9167e9c2b289aa32 [704807b]
fef3424a2dbfe91	2019-03-25 11:24:34	PFCD-5329	PFCD-5329 api.trace() of Maps or Lists values not working in Test Formula tracer
e25a479e340db7b	2019-03-25 10:07:06	PFCD-5332	PFCD-5332 Attachments > File Upload: Fix text/translation
486a12811e6e718	2019-03-22 14:40:58	PFCD-5334	PFCD-5334 Avoid 404 for missing highcharts.js.map and other *.map files
e4b94c3ffe99ca8	2019-03-27 12:10:17	PFCD-5340	PFCD-5340 Simple Filter on date fields - error when using #, !#, etc

a8ceb4fd642c7f6	2019-03-26 09:57:33	PFCD-5341	Revert "PFCD-5341 new Date().toTimestamp() not working after upgrade to El Presidente" This reverts commit e749b211d9650721bd3da3d87806c52c45a65ead.
e749b211d965072	2019-03-26 09:49:26	PFCD-5341	PFCD-5341 new Date().toTimestamp() not working after upgrade to El Presidente
f63c5b02806e079	2019-03-28 09:58:09	PFCD-5342	PFCD-5342 sku as default sort field breaks indey usage in Matrix* entities
77e4f7ee4e438dc	2019-03-27 14:20:32	PFCD-5345	PFCD-5345 Do not try saving api.global if too big for DB
c8e957a25353a5c	2019-03-27 12:47:51	PFCD-5345	PFCD-5345 Do not try saving api.global if too big for DB
3259eed15631fae	2019-04-01 10:42:44	PFCD-5352	PFCD-5352 Money format fields are displayed as real numbers without currency symbol when configured using header logics
a8bd9b0daeebe8d	2019-04-01 10:37:59	PFCD-5352	PFCD-5352 Money format fields are displayed as real numbers without currency symbol when configured using header logics
d3adf50e0cc3d7e	2019-04-01 10:26:21	PFCD-5352	PFCD-5352 Money format fields are displayed as real numbers without currency symbol when configured using header logics
c81f114f36b3900	2019-03-28 22:28:17	PFCD-5353	PFCD-5353: PO : add preferences option to Policy Records data grid
54287cd8ec8bc45	2019-04-02 14:58:39	PFCD-5354	PFCD-5354 Avoid an error when opening the PA Simulations page ... also removed one console log message
c3274c230601daa	2019-04-02 14:56:32	PFCD-5354	PFCD-5354 Avoid two SGWT warnings when opening the PA Simulations page
47e94a2837c3f0d	2019-04-02 12:28:29	PFCD-5354	PFCD-5354 restore field picker checkboxes anywhere but in Rollups When we decided to remove the multi-select checkboxes in the Rollups editor page, we also dropped them from PA Simulations and some places in PO. Now only Rollups have multi-select disabled, and hence no checkboxes. Anywhere else it is again possible to select all the fields at once.
b0fd8aff5629b69	2019-04-16 11:42:01	PFCD-5357	PFCD-5357 Fix permission issue when trying to list users in the add watcher dialog without having USER_FETCH permission
38dda7207c7c7c0	2019-04-16 11:42:01	PFCD-5357	PFCD-5357 UserIT.groovy fix and optimization. Introduced flag for permission skipping instead of path whitelisting.
e3dd759f2790122	2019-04-01 08:58:37	PFCD-5360	PFCD-5360 Align user permission checking in backend
ea5a980dcb0acc8	2019-04-01 08:56:10	PFCD-5360	PFCD-5360 Align user permission checking in backend
bfe61da53390e39	2019-04-02 12:24:50	PFCD-5361	PFCD-5361 Pipe through mail.* system properties to mail subsystem
052dd660bec8bb8	2019-04-03 08:21:15	PFCD-5368	PFCD-5368 Fix getAllAttributesMap (was only working for PLI; perf optimization)
1ac451eb8c6b768	2019-04-03 11:59:44	PFCD-5369	PFCD-5369 fix items
0ceb9973bc73f6d	2019-04-03 08:41:04	PFCD-5369	PFCD-5369 changes to always editable items' inputs are not correctly persisted
a4f1b8c68606807	2019-04-11 13:36:03	PFCD-5371	PFCD-5371 Fixing IT
06876bdd77c7f66	2019-04-05 12:20:45	PFCD-5371	PFCD-5371 Simulation Matrix conditional filtering
d725f3362ab48eb	2019-04-03 11:26:39	PFCD-5371	PFCD-5371 Make matrix logic result comparison a string-2-string comparison
ee6d4676cd5bbc2	2019-04-09 20:57:48	PFCD-5387	PFCD-5387: PA: formatted PA numbers to be aligned RIGHT (was LEFT)
46529615ea9b6da	2019-04-12 07:43:58	PFCD-5388	PFCD-5388 Restore DataLoadType.MR_CALCULATION and add handling We need it as it's the ordinal values of the enum that are persisted
08a8661262f17ec	2019-04-11 17:06:41	PFCD-5388	PFCD-5388 Remove unused DataLoadType.MR_CALCULATION ...which was not handled in DataLoadType.match() and DataLoadType.inverseMatch()
fdfca9de9162b30	2019-04-11 16:32:05	PFCD-5388	PFCD-5388 Int test for unknown DataLoadType in GetDataLoads

da5ad67592315a9	2019-04-11 16:23:43	PFCD-5388	PFCD-5388 Improve GetDataLoads error handling for unknown DataLoadType
10ff3c5426a757c	2019-04-11 16:21:14	PFCD-5388	PFCD-5388 Avoid StackOverflow in DataLoadType.toString() Was happening if inverseMatch() failed because of unhandled enum value.
6cb2c5e6609f1c1	2019-04-11 08:17:05	PFCD-5392	PFCD-5392 no limit on datasource options fetch
64cb44b5b87b5a6	2019-03-15 11:18:18	PFSFDCQC-2352	PFSFDCQC-2352: translations
26d3b7a0d29cc7e	2019-03-11 10:23:49	PFSFDCQC-2520	feat(pa): add translations (PFSFDCQC-2520)
4317ce1f19e062f	2019-04-04 10:40:34	PFSFDCQC-2691	PFSFDCQC-2691 merging into core localisations
9f4a4fa2a5559e7	2019-03-07 17:50:06	PM-284	Revert "PM-284 Alternative to api.getElement()" This reverts commit f455a305
d725f3362ab48eb	2019-04-03 11:26:39	string-2	PFCD-5371 Make matrix logic result comparison a string-2-string comparison

Salty Dog 3.5 Changelog

Commit	Date	Issue	Message
53a86e965203a06	2018-12-13 11:17:46	CVE-2018	CVE-2018-8088 we are not using EventData
e857d36d25abc16	2018-11-26 16:26:28	PFCD-3265	PFCD-3265: DatamartRowSet.next() skips the first row and points directly to the second row : IT "DatamartRowSet allows access to all Source rows"
ad4825968bc24fe	2018-11-05 16:04:16	PFCD-3415	PFCD-4175: PA DataLoads UI oddities PFCD-3415: Dataload "Detail" screen must show properly the current setting fixed
48884abc3cd6aff	2018-11-05 16:00:33	PFCD-3415	PFCD-4175: PA DataLoads UI oddities PFCD-3415: Dataload "Detail" screen must show properly the current setting fixed
f47da0fc2f685b6	2018-11-03 23:14:03	PFCD-3753	PFCD-3753: join query fix; minor change and lets QueryIT run succesfully
1f3c74be95f6f68	2018-11-06 11:37:30	PFCD-4175	PFCD-4175: PA DataLoads UI oddities: JSTgrid refresh delay set to 0
ad4825968bc24fe	2018-11-05 16:04:16	PFCD-4175	PFCD-4175: PA DataLoads UI oddities PFCD-3415: Dataload "Detail" screen must show properly the current setting fixed
48884abc3cd6aff	2018-11-05 16:00:33	PFCD-4175	PFCD-4175: PA DataLoads UI oddities PFCD-3415: Dataload "Detail" screen must show properly the current setting fixed
5c97d8d32807a72	2018-11-09 17:41:22	PFCD-4242	PFCD-4242 TimeSeriesScatter: setting min X and/or max X breaks the chart Providing Highcharts with the getMinX() and getMax() values didn't work, as they return the long timestamp value, whereas the x value of the data is the timestamp's *index* (0 to n-1). But since the query is updated according to the extrema, all data points are in the requested range anyway, so there is no need for setting Highcharts min/max values at all.
d9f00656d452ab6	2018-11-06 14:52:12	PFCD-4261	PFCD-4261: Incorporating reviewer's comment
c1ca0defb662a6d	2018-11-06 14:02:17	PFCD-4261	PFCD-4261: Showing just those workflow template types that are used for emails
6b52ee275a1db89	2018-11-16 11:47:08	PFCD-4292	PFCD-4292 hibernate 5.3 marks tx where staleexception happens as rollbackOnly, ensure that commit is not called eventually
a9a1089e659a8c8	2018-11-15 11:23:31	PFCD-4292	PFCD-4292 do not swallow stale object exceptions
120160f9e0e2f9a	2018-10-22 13:36:43	PFCD-4325	PFCD-4325 Fix incorrect filter
a1ce797df5b6668	2018-11-05 20:00:44	PFCD-4556	PFCD-4556: Editing datasource value creates a new dummy record in the UI fixed
5e0be8550a553e4	2018-10-17 13:26:39	PFCD-4570	PFCD-4570 Changing case of element name in Workflow logic throws unexpected exception
ab7d7401f9b11d5	2018-10-17 14:14:28	PFCD-4590	PFCD-4590 Fix misleading Query.toString() Some filters and many other settings were missing, because we were not overriding toString().

6cc8cb003878f70	2018-10-31 10:12:21	PFCD-4608	PFCD-4608: PDF download available even if CloudConvert is disabled
fd4d7e5e90dc315	2018-10-16 12:35:07	PFCD-4608	PFCD-4608 Store secure tokens in keystore
8437711b1068bcd	2018-11-22 14:11:21	PFCD-4626	PFCD-4626: load docx feature flag during login
5ad23391f630389	2018-11-22 09:06:51	PFCD-4626	PFCD-4626: Download DOCx disabled by default as servers are not available yet. You can enable this feature per partition by adding Advance Configuration Option: allowDocxConversion=true
54b3089e731067f	2018-11-06 12:54:56	PFCD-4626	PFCD-4626: default templating object passed with LineItems for CT,RA,Q
10b7244168e94d8	2018-11-06 12:53:11	PFCD-4626	PFCD-4626: Added default urls for templating and conversion
937fb8d5da2f529	2018-11-05 14:50:22	PFCD-4626	PFCD-4626: Merge branch 'master' into PFCD-4626
16c7b250c45363f	2018-09-24 13:49:24	PFCD-4626	PFCD-4626: Added Docx download possibility for entities that support templating
2b541790ffb95ed	2018-11-26 20:28:41	PFCD-4638	PFCD-4638: fields grid not always cleared when changing FC
2d057220d7174ed	2018-11-22 18:45:17	PFCD-4638	PFCD-4638: Number formatting of DS/DM fields
efdcaa7baaecb1c	2018-11-21 22:09:14	PFCD-4638	PFCD-4638: Number formatting of DS/DM fields
8fac13809869d42	2018-10-03 13:06:30	PFCD-4657	PFCD-4657 Price Grids selected folder is not kept when switch
89fed74a4730fc7	2018-10-02 07:36:46	PFCD-4658	PFCD-4658 Possible to select customer outside of user-specific filter
b8a48bcb2e78234	2018-11-03 19:08:02	PFCD-4660	PFCD-4660; merge fix
ff652e82b8c04a4	2018-11-03 18:13:47	PFCD-4660	PFCD-4660; also migrating' PA ITs from develop as 1) better and 2) there are issues with the previous ITs in terms of TX isolation (may need a few commits...)
77bc70bb845927f	2018-11-03 17:39:15	PFCD-4660	PFCD-4660; removing duplicate db migr scripts
6b0804d582fceb0	2018-10-21 17:53:01	PFCD-4660	PFCD-4660 PA query cache
f358fd0334ebbd1	2018-10-02 21:44:47	PFCD-4660	PFCD-4660: tweaks
8835463f587691a	2018-10-02 21:35:09	PFCD-4660	PFCD-4660: tweaks
42c916cad253528	2018-10-02 20:31:32	PFCD-4660	PFCD-4660: cache invalidation
9d80046a1c9e959	2018-09-30 19:53:07	PFCD-4660	PFCD-4660-pa-move-the-query-cache-index
df46ca88da2a2bb	2018-09-27 14:42:41	PFCD-4660	PFCD-4660-pa-move-the-query-cache-index
89ceb4fceaec276	2018-10-01 13:09:44	PFCD-4666	PFCD-4666 fix local filtering on Users' boolean fields
eb18b9a0e4b5b81	2018-10-16 07:10:00	PFCD-4681	PFCD-4681 Label for the field "key2" is not displayed in Matrix LPG / PL
cb12abd95b161b4	2018-10-09 12:07:47	PFCD-4683	PFCD-4683 DCR based on matrix PP: values where key is number aren't displayed
23e4050bfcbed55	2018-11-05 23:26:43	PFCD-4685	PFCD-4685: QueryIT: relax quantiles margin of error (actuals use interpolation while IT does not by design)
b158f297872e3ca	2018-10-30 21:22:58	PFCD-4685	PFCD-4685: PA quantiles calculation fixed
17c80fa358dbd86	2018-10-11 07:52:17	PFCD-4692	PFCD-4692 RA, RR: It isn't possible to mass-approve when your user is WF admin

24bcf168cf740c4	2018-10-24 06:56:34	PFCD-4697	PFCD-4697 Add filter reconstruction in runSummaryQuery
d144f677ee1304b	2018-10-24 06:54:35	PFCD-4697	PFCD-4697 Add filter reconstruction in runSummaryQuery
bf8887539691aee	2018-10-22 07:39:53	PFCD-4697	PFCD-4697 Add runSummaryQuery
86ba70ab0b563db	2019-01-08 11:02:01	PFCD-4701	PFCD-4701 DateTime(approval date) column is shown in wrong format after manual approval.
5806a28befc9564	2018-10-19 11:53:31	PFCD-4701	PFCD-4701 DateTime column is shown in wrong format after manual approval.
82c05aa4a77ae5a	2018-11-06 14:07:15	PFCD-4711	PFCD-4711: Removing no longer valid notes.
f4c9d86b6cf40f7	2018-10-31 15:37:21	PFCD-4711	PFCD-4711: Possibility to forward attachments when submitting without saving or reopening CLIC
1fdde540d3b4600	2018-11-02 09:47:34	PFCD-4717	PFCD-4717 GP: index creation bug in DM loader
f6db6ef23fc20a6	2018-10-26 10:23:48	PFCD-4717	PFCD-4717: branch had Rollup materialization code - was easier to add all of that rather than pull it out.
7e1e12e7636983c	2018-10-26 10:16:25	PFCD-4717	PFCD-4717: branch had Rollup materialization code - was easier to add all of that rather than pull it out.
17ecb81afea150c	2018-10-21 17:45:40	PFCD-4717	PFCD-4717: missed a bit while merging
21567b185f4ef7a	2018-10-21 16:46:15	PFCD-4717	PFCD-4717 PA Greenplum adaptations - merging in changes from Bosch PoC
545f4feb2ade444	2018-10-18 18:20:18	PFCD-4717	PFCD-4717 PA Greenplum adaptations...
eacef07ff440c39	2018-10-18 14:23:06	PFCD-4717	PFCD-4717 PA Greenplum adaptations...
6689c4b5f889fd9	2018-10-18 13:28:28	PFCD-4717	PFCD-4717 PA Greenplum adaptations...
34e6c954021fe13	2018-10-14 22:40:04	PFCD-4717	PFCD-4717 PA Greenplum adaptations...
a16e4c4bb7d5379	2018-10-20 11:01:30	PFCD-4718	PFCD-4718 -pa-query-throttle test must be adjusted according to the latest change; marking it @Ignored until fixed
98c89f437a36ca0	2018-10-15 07:49:46	PFCD-4719	PFCD-4719 Quote/Contract mass update, missing calculation messages
2f71ac9d40e8365	2018-10-16 12:53:29	PFCD-4725	PFCD-4725 modify createdBy/modifiedBy columns also on start panels
cf3a3452974ae61	2018-11-05 21:15:55	PFCD-4726	PFCD-4726: PA jobs fails on unclear message "Last block not stored, possibly due to out-of-memory Fixed
1b8da41b646d8e6	2018-10-24 13:44:48	PFCD-4726	Revert "PFCD-4726: trying H2 upgrade 1.4.196 -> 197" This reverts commit 2c278edb3f4611ef6cb733c302a040cbb6c2de8e.
c6112fc058ad17d	2018-10-23 09:46:45	PFCD-4726	Revert "PFCD-4726: trying H2 upgrade 1.4.196 -> 197" This reverts commit 2c278edb3f4611ef6cb733c302a040cbb6c2de8e.
2c278edb3f4611e	2018-10-23 09:38:02	PFCD-4726	PFCD-4726: trying H2 upgrade 1.4.196 -> 197
4a1e85e028a95be	2018-11-08 10:00:57	PFCD-4729	PFCD-4833 2nd pass calculation in distributed mode puts different values in api.batchInfo Some major refactoring, adding in server side elements of PFCD-4729 for simplicity
ff9ef445f0fd7cb	2018-10-24 19:35:24	PFCD-4730	PFCD-4730 Mass Edit does not create ITEM_UPDATE_PR event
1b8ba26327947de	2018-10-22 10:39:00	PFCD-4730	PFCD-4730 Mass Edit does not create ITEM_UPDATE_PR event
ceb741fc5fbc830	2018-10-24 10:05:40	PFCD-4733	PFCD-4733 Conversion of matrix pricegrid to pricelist doesn't work correctly
69f9221e7d39672	2018-11-01 15:38:35	PFCD-4737	PFCD-4737 PA Charts: Make sure CurrencySelectItem title doesn't get lost

1ca5074f28cada6	2018-10-23 07:35:25	PFCD-4739	PFCD-4739 App freeze in View Workflow History
983e482cc15abaa	2018-10-23 08:07:50	PFCD-4743	PFCD-4743: PA stream query result cache file not always deleted on close() - fixed
28a48d0b5409334	2018-10-22 18:36:11	PFCD-4743	PFCD-4743: PA stream query result cache file not always deleted on close() - fixed
4faf32268620209	2018-10-22 20:55:56	PFCD-4744	PFCD-4744 updates through reflection might lead to not persisting entity
2fccbf602404b6f	2018-10-23 06:37:55	PFCD-4745	PFCD-4745 XSS issue in custom online help
32598081beee70a	2018-10-23 08:50:47	PFCD-4747	PFCD-4747 Option to disable activity metric collection on a cluster level
b70bc6ebcf83b4e	2018-10-25 12:14:20	PFCD-4748	PFCD-4748 Create a method to specify multiple creators in a single document creation workflow step
c7796f9f58de1b2	2018-10-23 14:45:18	PFCD-4749	PFCD-4749 SAML SSO : Allow 5 seconds grace period on NotBefore date check to allow for millisecond clock skew
e8aca3bf5912925	2018-10-24 14:07:30	PFCD-4751	PFCD-4751: PO: cannot lookup record when wfStatus is NULL
0e99719199421ec	2018-10-24 12:18:00	PFCD-4753	PFCD-4753 api.find() do not work for GLOBAL PX after SaltyDog upgrade
fe5feb4e5d2dddb	2018-11-05 14:12:32	PFCD-4754	PFCD-4754 Fix IT
b03190120c28c2d	2018-10-31 12:16:26	PFCD-4754	PFCD-4754 Fix equals issue, remove filtering by state and add pre script for clearing wrong state
d1f935f7bec9375	2018-10-30 14:56:19	PFCD-4754	PFCD-4754 Clear active users when invalidated
261da16df192295	2018-10-26 11:35:11	PFCD-4754	PFCD-4754 My Pending Approval shows Approved and Denied in Salty Dog
3e8c0f6f7497f84	2018-11-08 18:19:46	PFCD-4755	PFCD-4755 Rollups Editor: if no controller, then disable resizing Had missed the fact that depening on where the editor is used, some of the panes may be permanently hidden. If so, then it's best to remove all the resize bars. Also improved the visibility/sizing handler.if the rollups list does not appear (pane0)
2f7001eb5080498	2018-11-08 16:02:00	PFCD-4755	PFCD-4755 Rollups Editor: Fix showController toggle button Now that there are multiple resize bars and panes can be hidden with a single click, (un/)hiding the controller got a bit trickier.
5ec9eb82efff0e8	2018-11-08 15:39:33	PFCD-4755	PFCD-4755 Rollups Editor: Better handling when (un/)hiding the last pane Now that there are multiple resize bars and panes can be hidden with a single click, make sure the whole browser width is used when the last pane (data table) is hidden and restore its previous size when it's shown again.
66268f55943c777	2018-11-08 07:49:25	PFCD-4755	PFCD-4755 Rollups Editor: Add resize bar to every pane
7560caba9374fe4	2018-11-01 10:31:27	PFCD-4755	PFCD-4755 Rollups editor: replace name field by uneditable source field ... and also allow dragging the same field to the measures grid multiple times to ease editing.
971c260d0aee9e0	2018-11-01 09:40:57	PFCD-4755	PFCD-4755 Rollups Field Picker: Remove checkboxes for easier drag & drop With the selection checkboxes, drag & drop requires an additional click to select the item before it can be dragged.
edf5214dde5ff08	2018-10-26 14:55:32	PFCD-4755	PFCD-4755 Sort Rollups field picker by labels
f9e38af9f91fb24	2018-10-25 08:58:53	PFCD-4756	PFCD-4756 Hard-deleting workflows fails
6e57f0991e926ff	2018-11-01 14:13:25	PFCD-4763	PFCD-4763 fix submit case
6080bb7c81c7b98	2018-11-01 11:55:25	PFCD-4763	PFCD-4763 option to eventually consitent OBJREFs after save/newRev/copy of CT/RBA fix merge
833e75a790a16e1	2018-11-13 10:28:43	PFCD-4794	PFCD-4794: DMQueryCache is not invalidating on DS.ccy change
9a87225da83bbdf	2018-11-13 10:03:23	PFCD-4794	PFCD-4794: DMQueryCache is not invalidating on DS.ccy change
5b15075f093a6a6	2018-10-31 19:32:42	PFCD-4795	PFCD-4795 fix filtering

c24035fd0f867ea	2018-10-31 13:33:59	PFCD-4795	PFCD-4795 MassUpdate of configurator inputs
962b69c311ac99f	2018-11-01 12:55:37	PFCD-4799	PFCD-4799 - Chart panel on Home screen Hide empty lines and redundant Refresh button to make more space for the real content
95610bd70a976ba	2018-11-12 13:57:56	PFCD-4801	PFCD-4801 Message corrected
762b7aa59a4b6eb	2018-11-05 13:15:32	PFCD-4801	PFCD-4801 Issue with duplicity in DCR for MPL
32a2392d54788b5	2018-11-05 13:01:29	PFCD-4801	PFCD-4801 Issue with duplicity in DCR for MPL
6118757218f20fc	2018-11-07 07:16:34	PFCD-4802	PFCD-4802 Issue with duplicity in DCR for PX/CX
0ccef608e5deb72	2018-11-09 07:48:16	PFCD-4803	PFCD-4803 include business keys & transform key names based on attribute meta
f433d2e6506b38f	2018-11-09 07:43:38	PFCD-4803	PFCD-4803 include business keys & transform key names based on attribute meta
1db1cb22c8aa716	2018-11-07 11:26:11	PFCD-4803	PFCD-4803 Filter in DCR in case of duplicity
8acac917fad9518	2018-11-23 15:37:55	PFCD-4805	PFCD-4805 API doc enhancement
b988b5273fd0637	2018-11-30 16:46:37	PFCD-4807	PFCD-4807 PADataEditor: do not show data in the group summary rows... ... numbers are wrong if SUM is not the correct aggregation. PADataEditor is used in DF/DS/DM/Rollups data grids and a few other places.
0a122eefeede69f	2018-11-07 14:08:12	PFCD-4808	PFCD-4808 description of withReasons updated
02f895b84864761	2018-11-07 13:58:58	PFCD-4808	PFCD-4808 description of withReasons updated
10040f8327f1421	2018-11-02 11:53:57	PFCD-4813	PFCD-4813 wf email generation might fail even user has disabled wf emailing
d4f38c8ac2dd648	2018-11-13 07:56:44	PFCD-4816	PFCD-4816 Collapsed/Expanded folder state not preserved between quote copies
06e4dea1edfceb3	2018-11-23 07:34:52	PFCD-4823	PFCD-4823 Fix IT
788e12ec2a5a6dc	2018-11-12 07:58:30	PFCD-4823	PFCD-4823 LPG: Number of approved items doesn't match number on overview page
1c94158f84de914	2018-11-20 09:40:48	PFCD-4824	PFCD-4824 and the full localization update for 3.5.3 - CS, DE, ES, FR Approved-by: Tomas Lamr <tomas.lamr@pricefx.eu> Approved-by: Milan Boruvka <milan.boruvka@pricefx.eu>
1804e4106db75bc	2018-11-06 09:37:32	PFCD-4826	PFCD-4826 Export custom metrics for prometheus only once
3c4bb10e2e0f304	2018-11-07 07:40:08	PFCD-4827	PFCD-4827 Support of api.batchInfo() in matrix logic
964bd1d411d62a6	2018-11-13 12:29:33	PFCD-4830	PFCD-4830 Home > My ToDo List: Created By field is always empty
4a1e85e028a95be	2018-11-08 10:00:57	PFCD-4833	PFCD-4833 2nd pass calculation in distributed mode puts different values in api.batchInfo Some major refactoring, adding in server side elements of PFCD-4729 for simplicity
c5b71c40e30ff46	2018-12-12 13:20:58	PFCD-4836	PFCD-4836 Different date format used by Matrix or Charts and LPG
ddfa4b9af0c320a	2018-11-16 14:19:03	PFCD-4838	PFCD-4838 Keep filter and selected user preferences on rebate records table refresh
0160cd7489de274	2018-11-15 09:17:49	PFCD-4839	PFCD-4839: incorporating PR comments: commandContext.getUser()
392332d5cdf31da	2018-11-15 08:41:20	PFCD-4839	PFCD-4839: Improved messaging for mass approve/deny. fetchWfAdminAware fixed for users that are not authorized to the WF
e18b66611d6cc7a	2018-12-06 08:36:07	PFCD-4843	PFCD-4843: Cancel transaction in case of any Exception
766454ca1ed2b6e	2018-12-05 13:52:49	PFCD-4843	PFCD-4843: Preventing reset of flushMode after OptimisticLockException + error message typo fixed

28bd82948ec402c	2018-12-05 12:32:12	PFCD-4843	PFCD-4843: Preventing client to send more parallel requests for LPG mass approve/deny
3187c16960469f1	2018-12-03 14:34:31	PFCD-4843	PFCD-4843: Improved error message also when mass denying lpg items
7db39aca84d3ec8	2018-11-26 14:23:45	PFCD-4843	PFCD-4843: Flush before header state update
5fc93eae444cc02	2018-11-26 12:09:49	PFCD-4843	PFCD-4843: Merge branch 'master' into PFCD-4843
81f8e7be36e36a8	2018-11-26 10:39:18	PFCD-4843	PFCD-4843: Improved message window (group by error type, showing either details or records count) + server error logs
bf1df45780dcaa9	2018-11-22 13:11:48	PFCD-4843	PFCD-4843: More specific error reasons.
8c678e1d00b598d	2018-11-21 14:26:34	PFCD-4843	PFCD-4843: Handling exceptions for workflow operations to avoid rollbacks, manual flush mode, better displaying of errors + small fix
b83e0e810ad54c0	2018-11-11 22:48:04	PFCD-4846	PFCD-4846: PO: target margin lookup helper(s) - say from quote logic
551153516921c84	2018-11-12 14:46:18	PFCD-4847	PFCD-4847 - DL filter to be >= instead of >
39fb00d8d80f217	2018-11-10 11:27:26	PFCD-4848	PFCD-4848 Support partition blacklisting
cdc0307fc7139d8	2018-11-19 13:59:17	PFCD-4849	PFCD-4849 minor UI fix
0e46c3eddaabbdd	2018-11-12 14:32:52	PFCD-4849	PFCD-4849 LPG calculation crashes if dynamic filter returns 0 records
8acfc3a8f5d52b4	2018-11-12 11:42:28	PFCD-4850	PFCD-4850: PA Rollup: source query not updated when saved in the UI + limit Rollups returned as Template DSs
20e02ab9b425ceb	2018-12-04 09:53:45	PFCD-4851	PFCD-4851 Fix dashboard preferences loading
bcd7612128215ee	2018-11-12 17:32:10	PFCD-4851	PFCD-4851 Dashboards: disable Load Prefs menu item if there are no prefs
cea97e346021f40	2018-11-29 08:02:30	PFCD-4852	PFCD-4852 PA Charts: safer use of isc.Time.getDefaultDisplayTimezone() For some reason this call sometimes fails (in Safari and only when the javascript console is closed) with this error: '((_3?-1:1)*H).stringify' is undefined Now making sure to call only when needed, and within its own try/catch clause.
5a38094a728fe1f	2018-11-29 07:58:43	PFCD-4852	PFCD-4852 PA Charts: One more null check in ChartEditor.loadSeries() ... in case HighchartWidget.attach() failed
e5a41ed19224591	2018-11-29 07:56:16	PFCD-4852	PFCD-4852 PA Charts: Better logging if HighchartWidget.attach fails
fb38acb65ae6275	2018-12-06 13:55:33	PFCD-4853	PFCD-4853 Empty SSOSELECTOR (i.e. default) confuses SSO relogin
5487b5ba8d90bba	2018-11-19 07:56:01	PFCD-4855	PFCD-4855: IT to unit test
6c2a7e324152b12	2018-11-15 15:04:26	PFCD-4855	PFCD-4855: fixed typo
bcd01fd4ce5df61	2018-11-15 14:56:50	PFCD-4855	PFCD-4855: NPE when fetching publishing templates + IT
4054d3f108bdba6	2018-11-15 10:46:00	PFCD-4866	PFCD-4866 Waterfall Comparison Chart: code tidying ... addressing one code review comment, and then some more
2aeb2863b973634	2018-11-15 08:13:33	PFCD-4866	PFCD-4866 Waterfall Comparison Chart: categories lost on reload/refresh When loading a saved chart from an already open chart editor, or when refreshing a dashboard, the categories on the X axis change to numbers.
73d5320819c308c	2018-11-17 06:14:38	PFCD-4880	PFCD-4880 Perf tweak: Make api.product("sku") not do a DB round trip
91b1bbbf5d9b3ee	2018-11-29 14:47:53	PFCD-4881	PFCD-4881 PA DataTable: "initial data tab grouping" option UI support Do group the DataTable according to the new ChartBuilder API option, default is ungrouped. There is no new checkbox so set this in the chart controller yet. We may add that in the next release.
035cf7b311bb468	2018-11-29 14:38:39	PFCD-4881	PFCD-4881 DataTable ChartBuilder API: "initial data tab grouping" option new API call to set whether a PA Data Table should initially be displayed as grouped (tree view)

03b5914ba2ffd75	2018-12-17 18:14:48	PFCD-4890	PFCD-4890 Missing import
202233b58d12225	2018-12-17 16:25:04	PFCD-4890	PFCD-4890
7484e90db5e209c	2018-12-17 16:24:33	PFCD-4890	PFCD-4890
e8cd3e76a44bac2	2018-12-17 16:07:51	PFCD-4890	PFCD-4890: removing requirement for a PO lincence for using regression in Matrix2D
dcc75cc8f11a55f	2018-12-17 13:26:14	PFCD-4890	Revert "PFCD-4890: Support Non-Negative Least Squares in Groovy" This reverts commit 3fb5780a55b71290027fa3adac22db4aba2a0d50.
1a70112bfc15e85	2018-12-17 13:25:06	PFCD-4890	PFCD-4890: Support Non-Negative Least Squares in Groovy
3fb5780a55b7129	2018-12-14 01:11:15	PFCD-4890	PFCD-4890: Support Non-Negative Least Squares in Groovy
2f118f2aa0f717b	2018-11-21 14:19:41	PFCD-4891	PFCD-4891: PA: create 'legacy' Product & Customer DLs also in SD
0fd889bb9bc5f00	2019-01-18 17:09:38	PFCD-4893	PFCD-4893 version conflict after (re)naming a rollup
872795a1ea80eab	2019-01-17 13:15:11	PFCD-4893	PFCD-4893 unable to save a new rollup Was due to bad way of trying to show the new grid record as selected. Now following Isomorphic's recommendations about unsaved records.
16624dd53b1fd0b	2018-12-10 15:44:36	PFCD-4893	PFCD-4893 Rollups: new rollups are now empty, and added a copy button... ... and improved selection handling, no need for the edit button anymore
f2270328c7a54d5	2018-11-27 13:14:17	PFCD-4896	PFCD-4896 Mass submit is not working when default calculation is not set.
9d820cea46ddfef	2018-11-26 13:28:30	PFCD-4897	PFCD-4897 Link from default msg template "rebaterecord_approvalRequired " is not working correctly
0a720873e968ae4	2018-11-22 16:43:59	PFCD-4899	PFCD-4899 â¤ IT
ee577f90f528b73	2018-11-21 15:15:09	PFCD-4899	PFCD-4899 simplify
6d23140c31a272d	2018-11-21 14:41:02	PFCD-4899	PFCD-4899 overwrite filters that reaches oracle in limit also when set through setValue
bfe35fbf9b1fe29	2018-11-22 16:25:57	PFCD-4902	PFCD-4902 qmu: excpetion on cg filter matching multiple quotes
ac09eb43dd37875	2018-11-22 19:09:01	PFCD-4903	PFCD-4903 mass update, messages are not cleared on subsequent runs
f023a41784ec425	2018-11-23 15:40:02	PFCD-4908	PFCD-4908 TargetDate returns null in matrix logic
046755047409759	2018-11-28 10:54:32	PFCD-4912	PFCD-4912 refresh inline configurator (outer) model also when "noRefresh" is set inside
2be54b2452905d9	2018-11-28 06:31:04	PFCD-4912	PFCD-4912 Add timer to give inline config time to finish its processing - increasing timer value a bit
190156788e465aa	2018-11-27 14:02:01	PFCD-4912	PFCD-4912 Add timer to give inline config time to finish its processing
a31bb26d2d6df93	2018-11-29 12:16:28	PFCD-4913	PFCD-4913 The value of DMDIMFILTER entry, used in inline configurator, gets erased
16fedcdb7a1575e	2019-01-08 12:14:06	PFCD-4915	PFCD-4915 approval through comment form
19d32490e0daac1	2018-11-05 09:42:19	PFCD-4921	PFCD-4921: Hover-over title of configuration renamed to administration
024b0aa5d01bc79	2018-11-29 12:38:05	PFCD-4922	PFCD-4922 - Documentation improved for Public API - api.function()
1e88a08a5e4871f	2018-12-03 09:59:16	PFCD-4927	PFCD-4927: Code format fix
4abf8c68c0288d0	2018-12-03 09:54:29	PFCD-4927	PFCD-4927: Docx button enabled by default. However you can turn off "Docx download" via disabledDocxDownload app property

4f87aa49c2d4d7b	2018-12-05 11:58:56	PFCD-4928	PFCD-4928 OBJREFS filter broken in C details
86413deda60679b	2018-12-03 08:15:02	PFCD-4928	PFCD-4928 C/P <-> RBA ObjRefs broken
d74c60ee608b040	2018-12-04 07:42:30	PFCD-4930	PFCD-4930 api.getBatchInfo() doesn't return secondary key during secondpass in priceList
15b6d91cb4e7483	2018-12-10 09:25:06	PFCD-4937	PFCD-4956 - Unable to open dashboard (possible regression) - Fixed NPE caused by PFCD-4937
d301a5e5397a3e3	2018-12-06 12:37:20	PFCD-4941	PFCD-4941 - API call api.options with labels - added IT
7126104cefde913	2018-12-06 12:22:46	PFCD-4941	PFCD-4941 - API call api.options(entryName, options, Map<String,Object>) is not working as explained in the documentation
1150b5871a55986	2018-12-16 10:03:32	PFCD-4942	H2 version inc to address failing QueryIT and probably also PFCD-4942: PA QueryCache closed after previous exception (not reporducable locally on 197).
5c39901289746b4	2018-12-06 11:06:08	PFCD-4942	PFCD-4942: use ConcurrentMap for storing caches
3e123bab6e9d0ca	2018-12-05 15:47:08	PFCD-4942	PFCD-4942: PA QueryCache closed after previous exception
e866d033a8c705a	2018-12-05 15:19:16	PFCD-4942	PFCD-4942: PA QueryCache closed after previous exception
883caa89f2cd4a2	2018-12-06 09:00:52	PFCD-4944	PFCD-4944 clear UserContext before bg task is executed
f30094aa3c3266a	2018-12-06 11:26:37	PFCD-4946	PFCD-4946 Potential concurrency issue in ClusterService: to use ConcurrentMap for storing partition executorPools
312ad2df81ecbb7	2018-12-06 11:45:42	PFCD-4947	PFCD-4947 ensure lastUpdateDate is set to now on onSuperseded
79979de1aae5ba7	2018-12-10 19:04:15	PFCD-4948	PFCD-4948 fix npe and propagate formula errors to warning in SIM when doing updateI
c3236c5bd4eba86	2018-12-10 10:00:35	PFCD-4953	PFCD-4953 Matrix LPG: Deleting multiple records removes all instances of SKU's, not only marked ones
4449359543f1f19	2019-01-14 12:04:04	PFCD-4954	PFCD-4954: IT fix (re Rollup obj instantiation)
078473118e4ae96	2019-01-14 11:34:27	PFCD-4954	PFCD-4954: placeholder 'calc' fields not handled correctly in VIEW normalization (and IT was not marking them as calculated)
87ab68c95c9b77d	2018-12-08 11:58:44	PFCD-4954	PFCD-4954: GP in SD - drop unccessary param in bootstrap
7873e5d14e05762	2018-12-07 21:23:27	PFCD-4954	PFCD-4954: GP in SD
544a42ceb2ed737	2018-12-10 08:33:54	PFCD-4955	PFCD-4955 make inputs/outputs on CLItem eagerly fetched / WF activeUsers hibernate mapping PERF issue Before Salty Dog, setting FetchMode.LAZY had no effect and was ignored. All the inputs and outputs were eagerly fetched. Since now it is working, this causes perf issue when loading items, a db query for each item needs to be genearated.
15b6d91cb4e7483	2018-12-10 09:25:06	PFCD-4956	PFCD-4956 - Unable to open dashboard (possible regression) - Fixed NPE caused by PFCD-4937
2666f58a051cb48	2018-12-18 11:52:10	PFCD-4964	PFCD-4964 Date Format for Data Change Request in Excel Client has error
7acb17b277b5c55	2019-01-08 10:26:46	PFCD-4970	PFCD-4970 Missing import in PACalculationIT
13b396490016839	2018-12-17 16:04:09	PFCD-4970	PFCD-4970: typos in comments
55aa9d3649e59f3	2018-12-17 15:23:29	PFCD-4970	PFCD-4970: PA Calculation job gets stuck when feeder is slow to start emitting
7596b664a7fea40	2018-12-18 08:44:02	PFCD-4974	PFCD-4974 wf config broken
7dfa9c4aa3e59f4	2019-01-11 13:58:23	PFCD-4977	PFCD-4977 PB simulation of a matrix PL contains more (irrelevant) records

bcf141e67540d17	2019-01-11 13:40:02	PFCD-4977	PFCD-4977 PB simulation of a matrix PL contains more (irrelevant) records
070af9518a249a3	2018-12-20 13:20:40	PFCD-4978	PFCD-4978 Copy of PL sometimes doesn't work on granit-qa partition on staging-eu env
f8ef6efaa164e66	2018-12-20 11:36:12	PFCD-4981	PFCD-4981 <ref/> links fix
e34fa3e8634bd1d	2018-12-20 15:35:59	PFCD-4983	PFCD-4983 fix pg migration
de82d43511a2a19	2018-12-20 15:35:44	PFCD-4983	PFCD-4983 Proper data migration
a5435460519a313	2018-12-20 15:27:25	PFCD-4983	PFCD-4983 fix pg migration
338a4f637c263a1	2018-12-20 14:48:35	PFCD-4983	PFCD-4983 Proper data migration
5f625fde0498770	2018-12-20 13:12:39	PFCD-4983	PFCD-4983 fix hashCode contract in Workflow_activeUsers
4f10269544a9d2e	2018-12-20 13:01:49	PFCD-4983	PFCD-4983 fix hashCode contract in Workflow_activeUsers
e279551017ca975	2019-01-08 16:37:52	PFCD-4988	PFCD-4988: item groupBy covered in IT
01d93e1bf7a544d	2019-01-08 16:21:04	PFCD-4988	PFCD-4988: Added more item groupBy fields to SummaryView
4d462bf7f9fbbfa	2019-01-07 11:18:53	PFCD-4996	PFCD-4996 Simple unit test for Matrix2D.sortRows()
88a3b120f08af08	2019-01-07 09:25:55	PFCD-4996	PFCD-4996 Fix NPE in Matrix2D.sortRows() ... was due to setting type to null when MatrixUtil.sortRows() then does switch(type), now defaulting to "text".
cf4efd0be369578	2019-01-07 16:53:57	PFCD-5001	PFCD-5001 use 5.2.17.Final enhancer
4721fd8efe2f0b1	2019-01-07 16:30:00	PFCD-5001	PFCD-5001 hibernate5+ params perf issue
8eaa59a531f045e	2019-01-09 08:48:24	PFCD-5003	PFCD-5003 manual flush does not work with workflow
b3ea57c2c815b5b	2019-01-21 14:54:22	PFCD-5008	PFCD-5008 findRangeUsingIds does not apply am search callback on sort creation
4c5165ddc56b9c2	2019-01-14 09:44:56	PFCD-5016	PFCD-5016 untranslatabel splitted i18n props
395103ef20912e1	2019-01-22 08:37:15	PFCD-5017	PFCD-5017 Allow override of startDate and endDate for price records generated from line items - PromotionManager
86e134995cb550e	2019-01-16 14:02:08	PFCD-5024	PFCD-5024 Fix NPE in Heatmap (null/missing value in data point)
fdc52d04baca70a	2019-01-16 13:25:30	PFCD-5028	PFCD-5028 broken workflow delegation
66bab7cb08dba28	2019-01-18 11:59:24	PFCD-5038	PFCD-5038: DL Flush fails with NPE on DS with postprocessing logic on SD PR5 RC fixed, IT = DataSourceLoadIT."DS data remains aligned when flushing a modified DF"
743919a188a8263	2019-01-23 10:47:04	PFCD-5049	PFCD-5049 pricechange check might be triggered for resultPrice differences less than 0.00001
02a7c215ecee130	2019-01-23 14:53:24	PFCD-5051	PFCD-5051 fix datetime input
348a4e3fa9228e6	2019-01-23 11:14:06	PFCD-5063	PFCD-5063 set Highcharts decimal and thousands sep according to session
9b993f109fb4647	2019-01-30 12:04:22	PFCD-5087	PFCD-5087 more chicken fix
c2f4112e7da1fdf	2019-01-30 12:04:04	PFCD-5087	PFCD-5087 pgi maunal override - timestamp not saved in manualOverrides
a5392b2fe1fc6d8	2019-02-02 16:58:40	PFCD-5099	PFCD-5099: updated IR

a92e4bcc1806a48	2019-02-02 16:40:46	PFCD-5099	fix(PFCD-5099):: PA query error when DM has dim-DateTIme field (since GP mods)
ae21852b98641bf	2019-02-15 08:32:43	PFCD-5145	PFCD-5145 update/set of attributeExtensions though header logic is broken
1fca295814839f1	2019-02-21 14:02:30	PFCD-5158	PFCD-5158 User is no longer able to select on behalf of which approve/approval group he wants to approve
56439591897a50f	2019-02-27 10:23:55	PFCD-5172	fix PFCD-5172: Refresh of empty DM fails Note: ony effected STN, who have a legacy DSView
781f392c615750c	2019-02-28 15:12:20	PFCD-5209	PFCD-5209: PA Query Cache: index files keeps growing, result files not deleted Not found the root cause, but implemented obvious improvements and added a couple of params to be able to tweak mvstore setup.
9b3d12d814622fb	2019-02-25 07:57:24	PFSFDCQC-2271	PFSFDCQC-2271: Update translations from Unity

Cuba Libre 3.4 Changelog

Commit	Date	Issue	Message
c39a4ce3c40c184	2018-04-26 07:35:25	ORA-01427	PFCD-4296 Upgrade to Cuba Libre fails with ORA-01427
186c27b11bec688	2018-06-01 08:31:11	PCFD-4254	PCFD-4254 BubbleChart: add count aggregation to addtl measures Unit Test Would not have caught the issue from previous commit, but it's related nevertheless.
34cc0193cb5c622	2018-06-01 08:19:52	PCFD-4254	PCFD-4254 BubbleChart ChartBuilder Hint issue for COUNT aggregations When adding a bubble chart to a dashboard (or when requesting the ChartBuilder Hint code) the .addAdditionalMeasure() section was missing .withAggregation('COUNT').
5f3b53369a34add	2018-05-15 14:02:32	PFCD-2744	PFCD-2744 Tooltip fixes
2aecb65c35460cc	2018-04-24 10:33:19	PFCD-2921	Revert "PFCD-2921 diff in cuba libre debug" This reverts commit 736383cb0874b64d80fa35002e2f27190a2cfdb7.
736383cb0874b64	2018-04-24 10:20:14	PFCD-2921	PFCD-2921 diff in cuba libre debug
deaa9f5d86e2676	2018-05-11 13:00:13	PFCD-3089	PFCD-3089 Properly color added rows in Show Changes View
6b528166b327c6b	2018-06-12 14:32:53	PFCD-3213	PFCD-3213 persists line items details
05788c8fa32718a	2018-06-12 14:32:40	PFCD-3213	Revert "PFCD-3213 previous state has no lazy props" This reverts commit 32792caaa5db2ca0da9aa8d7fffdaebae60632a4.
32792caaa5db2ca	2018-06-12 12:46:42	PFCD-3213	PFCD-3213 previous state has no lazy props
8aa4900ea5a7d40	2018-08-08 13:19:26	PFCD-3377	PFCD-3377 PA: Fix FCSelectItem refresh
94780af80485a8d	2018-08-07 15:15:19	PFCD-3451	PFCD-3451 Fix dashboard recovery after execution error
6677581e8e9882d	2018-05-02 13:18:55	PFCD-3894	PFCD-3894 customize listgridfield instead of datasourcefield
a29e52b8f373459	2018-05-02 12:56:43	PFCD-3894	PFCD-3894 attribute meta, numerical restricted list fix
d88ef9d21c5af5d	2018-05-14 13:45:55	PFCD-3947	PFCD-3947 - Add French, Spanish and Czech as a standard language for UI Fixed one CS translation
cf3f67f13488344	2018-04-20 14:52:42	PFCD-3947	PFCD-3947 - Add French, Spanish and Czech as a standard language for UI Last version of Czech translation
9a88ef106b27e6d	2018-04-20 14:25:32	PFCD-3947	PFCD-3947 - Add French, Spanish and Czech as a standard language for UI cz -> cs (correct locale for Czech Republic is 'cs')
4237e29b2cb0b43	2018-04-26 13:15:44	PFCD-3991	PFCD-3991 RAT resolve version conflict issue
0b885e6875ce779	2018-04-20 08:19:51	PFCD-3991	PFCD-3991 All filled values from Attr. columns disappear

95eebd2517807e9	2018-04-12 12:29:17	PFCD-4005	PFCD-4005 JavaDoc update
f4d5ffaa320c11a	2018-07-25 16:31:30	PFCD-4041	PFCD-4041 Fix PAFieldType.format() so Integers have thousands separator
70fb76fa73b1ceb	2018-03-22 14:22:56	PFCD-4057	PFCD-4057 add double quotes around keys of FlexChart default templates Should be better for standard JSON parsers, although to be really compliant we should also find a way to wrap functions!
a2d176443739138	2018-06-05 10:08:34	PFCD-4067	PFCD-4067 show only one scroll bar
df2832e04ce566e	2018-05-07 07:18:26	PFCD-4136	PFCD-4136 Remove double escaping
1f07a112424c9c6	2018-06-08 13:36:00	PFCD-4151	PFCD-4151 After clicking on the preference manager drop down first time, the available value list pops up in the middle of the screen
d2bacd69f08fd54	2018-04-24 10:47:35	PFCD-4165	PFCD-4165 Revert restoring focus in configurator for some type of inputs(especially STRINGUSERENTRY etc. :( ) it caused issue in TATA
a6b09f7ff154ad5	2018-04-27 11:17:10	PFCD-4167	PFCD-4167 add api.isFullListCalculation()
2908fc04a0fb943	2018-04-26 10:44:22	PFCD-4167	PFCD-4167 add api.isFullListCalculation()
5792c9c8cee9cfb	2018-04-30 09:05:04	PFCD-4184	PFCD-4184 Extension tables - missing filters
36f9aba0570f9b6	2018-05-14 10:05:14	PFCD-4211	PFCD-4211 Allow job-individual shotgun mode thresholds from calc flows
9ee259babc82db7	2018-05-31 08:59:09	PFCD-4226	PFCD-4226 - integration test - incorporating comments from pull request
8cb54b75692e255	2018-05-31 08:22:31	PFCD-4226	PFCD-4226 - integration test - Last Updated By is stamped with the last approver/submitter.
c0ed9de42768735	2018-05-30 14:18:00	PFCD-4226	PFCD-4226 - view workflow button was added into DCR record. Last Updated By is stamped with the last approver/submitter.
ac26818be438d95	2018-05-30 14:17:22	PFCD-4226	PFCD-4226 - view workflow button was added into DCR record. Last Updated By is stamped with the last approver/submitter.
30764f1b1269b9b	2018-05-16 06:24:33	PFCD-4234	PFCD-4234 Not possible to get a percentage formatting in a Matrix PP with restricted values
fb1e1aec2bb98ad	2018-04-17 06:09:00	PFCD-4235	PFCD-4235 Whitelist rebate agreement payoutDate for updates
6e70700acca2940	2018-07-10 06:47:57	PFCD-4243	PFCD-4243 Value options for manual override of attributed result are not shown in LPG product detail view
f388fa85552abb9	2018-04-12 09:00:06	PFCD-4244	PFCD-4244 PX50 not supported for P/C filtering
9a89560e1394044	2018-04-13 06:11:09	PFCD-4245	PFCD-4245 Quotes overview page: Cannot filter by Superseded/Invalidated by field (also applies to RBA/CT)
e00b362cbf33b3d	2018-04-13 10:14:20	PFCD-4247	PFCD-4247 in case of two header level customer inputs, prefer default one for saving into PR
e5436fa97103223	2018-04-17 06:02:24	PFCD-4250	PFCD-4250 EC: required field doesn't work in CX EC Commit: 82496cf2f6aa1b4572429f65e67c25bc9ba8334f [82496cf]
9636c3251716866	2018-05-15 10:17:32	PFCD-4253	PFCD-4253 fix integration test
4c5a77b5c42bbff	2018-05-14 11:44:24	PFCD-4253	PFCD-4253 avoid resubmit of already submitted PGIs
99dcf1c0dda5c4e	2018-04-17 09:34:30	PFCD-4256	PFCD-4256 Don't allow to filter on BasePL id
3fdcbeb74c720dc	2018-05-04 14:23:44	PFCD-4257	PFCD-4257 Embedded dashboard inputs have "undefined" labels
864f9ef35d1801c	2018-05-04 14:21:40	PFCD-4257	PFCD-4257 Embedded dashboard controller sometimes expands after refresh
5f2c6d161ff2f21	2018-05-04 14:21:39	PFCD-4257	PFCD-4257 Dashboard UI: Avoid SGWT negative width warning Was logging "SpritedScrollbar:isc_IButton_41_hscroll:ignoring bad or negative width: -15"

98b1eb59bf06bb2	2018-04-24 12:35:54	PFCD-4257	PFCD-4257 Save dashboard controller's expanded/collapsed state
1722fcf1c63d8d0	2018-04-17 12:42:04	PFCD-4257	PFCD-4257 default to hiding embedded dashboard controller
73945e77080f4cd	2018-04-18 12:31:04	PFCD-4258	PFCD-4258 Quote Header should be invisible when all input fields under the group are hidden
3a3719515060d1f	2018-04-18 11:12:28	PFCD-4258	PFCD-4258 Quote Header should be invisible when all input fields under the group are hidden
cacd5eefa5947fe	2018-04-18 11:33:10	PFCD-4259	PFCD-4259 Strict CSRF check breaks XLS download
80d54502fc01945	2018-04-18 06:56:51	PFCD-4262	PFCD-4262 QC: On copying configured products, 'mandatory field' condition is not retained for Input Parameters
3859bbe783d9e55	2018-04-19 06:54:01	PFCD-4266	PFCD-4266 Fix "Cancel" of DataArchiving
ed26cd2a60923a1	2018-05-11 12:09:01	PFCD-4267	PFCD-4267 fix exports2
5085f2ba609bf41	2018-05-11 12:05:30	PFCD-4267	PFCD-4267 fix npe
a5d5767c07924b5	2018-04-24 07:38:55	PFCD-4267	PFCD-4267 Manually entered date filter on grid causes empty downloaded excel (cherry picked from commit fe73760)
b90298e4451b511	2018-04-19 08:08:23	PFCD-4268	PFCD-4268 Cannot create pricelist if another logic is failing on syntax check
b8baecf3875068a	2018-06-26 07:53:43	PFCD-4269	PFCD-4269 tiny perf optimization
769a1536bfc37cd	2018-06-26 04:54:54	PFCD-4269	PFCD-4269 - XPGIItem fix query split into several queries to respect MAX ORACLE/POSGRES params
4138aa118dd8076	2018-06-06 13:01:54	PFCD-4274	PFCD-4274 Cannot create Preference for Pricelist detail screen
62a7909d1501c85	2018-04-24 11:01:04	PFCD-4275	fix(PFCD-4275)
f11d8188c272b0f	2018-05-02 13:04:21	PFCD-4282	PFCD-4282 Add or update PP by EC requires BULKLOAD permission EC Commit: 4e918f6d15eeaca8cae4dfb1799c81189a560386 [4e918f6]
572122f82c472ef	2018-04-23 15:01:26	PFCD-4282	PFCD-4282 Add or update PP by EC requires BULKLOAD permission
6a133a214f13fba	2018-05-07 13:47:20	PFCD-4283	PFCD-4283 Also showing attribute technical names in header EC Commit: 9100c46da6287f216df0e463a86601f0d876d9be [9100c46]
cfa0c121d425af2	2018-04-24 07:33:02	PFCD-4284	PFCD-4284 is full list calculation not present in master
6603cfa2118d04a	2018-04-23 18:55:24	PFCD-4284	PFCD-4284 api.contextTypedId() in header logics
fe38cc6f5a4cc88	2018-05-07 07:32:20	PFCD-4286	PFCD-4286 Amend API doc
2122db437e27269	2018-05-02 09:57:53	PFCD-4286	PFCD-4286 formulamanager.executeFormula option to control output in raw json
781a3b39d95448c	2018-04-24 07:43:12	PFCD-4287	PFCD-4287 allow write from Creation Workflow Logic
540145240fda09f	2018-08-07 15:07:02	PFCD-4288	PFCD-4288 Backporting: Configurator: Fix merge
6ec8b4d0d4f53e7	2018-08-06 06:45:15	PFCD-4288	PFCD-4288 Backporting: Configurator: Show progress when being recalculated
3d000b6f21897dc	2018-04-26 16:59:38	PFCD-4293	PFCD-4293 UI: Cannot create new PL if P filter uses "equals" + rel dates Now ensuring that the same product filter is used for the products grid query and when fetching data for pricelist parameters.
441a3498331edf0	2018-05-09 12:44:51	PFCD-4294	PFCD-4294 Empty dowloaded excel when there is date filter on grid
c39a4ce3c40c184	2018-04-26 07:35:25	PFCD-4296	PFCD-4296 Upgrade to Cuba Libre fails with ORA-01427

77d4163c4b1902c	2018-04-26 07:44:46	PFCD-4297	PFCD-4297 Product Selector input in a quote hides other attributes
5ad2871f92ce74d	2018-04-26 08:02:55	PFCD-4299	PFCD-4299 Background tasks in remote schemas (with multiple partitions in it) are not triggered
5e0729c2dc6873f	2018-04-27 13:14:38	PFCD-4304	PFCD-4304 Fix for hidden inputs
ef0c07380f99ac6	2018-05-25 08:45:31	PFCD-4305	PFCD-4305 do not count inactive users
e587fff3b529ae0	2018-05-02 09:34:54	PFCD-4305	PFCD-4305 License counts are not accurate
c9f9bb66619231e	2018-05-02 12:36:39	PFCD-4307	PFCD-4307 Read-Only option on CustomerId/SKU column IN CX/PX doesn't work
9bab5f31faeffc8	2018-05-02 13:38:15	PFCD-4310	PFCD-4310 ResultMatrix: slowness when loading grid with 1000+ rows 'slow' meaning 30 seconds for 1200 rows ! Solved by avoiding loop on DS.addData(Record) as recommended in the method's javadoc.
3aa049bdcf7192e	2018-05-02 13:31:48	PFCD-4310	PFCD-4310 ResultMatrix: row click events are sometimes doubled Switching from SelectionUpdatedEvent to RecordClickEvent fixed this. Was probably due to AdvancedListGrid.performCellSingleSelection() doing an additional grid.deselectAllRecords() + grid.selectRecord() !
dec6fa3a0b7249f	2018-05-24 08:10:17	PFCD-4311	PFCD-4311 Tiny refactoring to use Optional.orElse() (thanks Tomas)
4619044f82aaa1b	2018-05-22 15:40:32	PFCD-4311	PFCD-4311 Dashboard controller: again tweak adaptive num of input cols
fafacd6c5501cce	2018-05-16 10:15:40	PFCD-4311	PFCD-4311 Dashboards: code tidying more obvious defaults and removed an unused label
2886910acc36552	2018-05-16 10:15:39	PFCD-4311	PFCD-4311 Dashboards: avoid an annoying SGWT warning when resizing
ebb42329d8703d5	2018-05-11 13:45:09	PFCD-4311	PFCD-4311 Dashboard controller: tweak adaptive number of input columns
93c411af8f3eec6	2018-05-03 12:01:44	PFCD-4312	PFCD-4312 limited Customer Input Filter for customer group picker entry
98304c5d6c1c54f	2018-05-16 10:54:01	PFCD-4314	PFCD-4314 translations in metas are not copied over
f096af2d3296ced	2018-05-23 12:48:19	PFCD-4315	PFCD-4315 - added check if private i18n messages are empty, added dialog to save also to private i18n messages when saving to partition wide i18n messages. (cherry picked from commit 8dfd56f)
5a0e66ba3163f55	2018-05-25 07:59:56	PFCD-4316	PFCD-4316 ui glitches
c18b23ac5729f59	2018-05-09 13:18:39	PFCD-4316	PFCD-4316 basket in product selector
c088f98aef90098	2018-05-07 13:06:08	PFCD-4318	PFCD-4318 Excel Client "Price List" selection - full name not visible EC Commit: 86de0a83bb18403d83d36515d9bb31be7b9672f6 [86de0a8]
678b8a548008668	2018-05-29 07:23:22	PFCD-4323	PFCD-4323 - removed logs
105aafe2a5176b5	2018-05-29 07:18:44	PFCD-4323	PFCD-4323 - applying pricelist configurtion filters whehn showing definition and making copy of pl
f6200c3520d5d6b	2018-05-09 06:45:14	PFCD-4324	PFCD-4324 Remove restore focus functionality
6deefa4fa842a35	2018-05-09 05:46:34	PFCD-4326	PFCD-4326 Option to disable screen refresh on change of input fields in configurator..
90470af4f498d55	2018-05-10 07:39:47	PFCD-4327	PFCD-4327 no filtering and sort support for input in product selector
5c81b1a5bb21052	2018-05-10 06:28:53	PFCD-4329	PFCD-4329 - Quote cannot be converted if the Expiry Date is today Fixed date comparison to ignore time.
f91aaa887e3fd6f	2018-05-11 12:05:40	PFCD-4334	PFCD-4334 pdf and xls links are swapped in rba console
0ab1d56d7d08e48	2018-05-25 12:52:02	PFCD-4336	PFCD-4336: extending test coverage

e64956454681aed	2018-05-25 12:46:40	PFCD-4336	PFCD-4336: extending test coverage
2ea6f532506657d	2018-05-25 12:46:15	PFCD-4336	PFCD-4336: using calculated field as group by...
3c5a93852a950dc	2018-05-25 12:45:45	PFCD-4336	PFCD-4336: using calculated field as group by...
56c00905db80730	2018-05-23 19:09:03	PFCD-4336	PFCD-4336: fix calc field dereferencing in sql resolution
f8d10c16ebd25d7	2018-05-23 18:57:51	PFCD-4336	PFCD-4336: fix calc field dereferencing in sql resolution
2a344cf49bf32bc	2018-05-21 08:23:07	PFCD-4336	PFCD-4336: fix case sensitivty in-DM placeholder field resolution
53eeba02bc9bbe5	2018-05-21 08:01:26	PFCD-4336	PFCD-4336: fix case sensitivty in-DM placeholder field resolution
f7c6807ec243f72	2018-05-18 09:14:13	PFCD-4336	PFCD-4336 DatamartIT typos
aa81a016d593ac5	2018-05-18 09:14:01	PFCD-4336	PFCD-4336 DatamartIT cosmetic changes (like missing Unroll annotation)
07be321bf256fd8	2018-05-18 09:13:50	PFCD-4336	PFCD-4336: field calculations in DM are using unconverted amounts (money & quantity)
431299b2b6c65dc	2018-05-18 09:13:33	PFCD-4336	PFCD-4336: field calculations in DM are using unconverted amounts (money & quantity)
622992fa3fbeb66	2018-05-17 14:52:34	PFCD-4336	PFCD-4336 DatamartIT typos
67ba6296a930a7e	2018-05-17 14:18:24	PFCD-4336	PFCD-4336 DatamartIT cosmetic changes (like missing Unroll annotation)
d6efe649e2d1f8e	2018-05-17 11:02:23	PFCD-4336	PFCD-4336: field calculations in DM are using unconverted amounts (money & quantity)
b21f0e290bad03d	2018-05-16 15:16:49	PFCD-4336	PFCD-4336: field calculations in DM are using unconverted amounts (money & quantity)
4698e77e7a3ff43	2018-05-13 11:36:43	PFCD-4336	PFCD-4336: field calculations in DM are using unconverted amounts (money & quantity)
d26b82bb3e6feac	2018-05-22 11:57:53	PFCD-4338	PFCD-4338 Do not automatically move focus to empty required fields
13b7c29db726b7f	2018-05-18 14:18:45	PFCD-4341	PFCD-4341 fix pending creation workflow when run from save
ba65ec44a1a0eaf	2018-05-18 12:47:56	PFCD-4341	PFCD-4341 fix url
549145e151c7b52	2018-05-17 18:57:12	PFCD-4341	PFCD-4341 fix back in cwf
516a7d41f0bdf8f	2018-05-16 08:28:37	PFCD-4341	PFCD-4341 Ability to stop with error in creation workflow logic
b3cddac66329e4e	2018-05-28 16:33:13	PFCD-4345	PFCD-4345 reset customerId also on save before cwf
d1278c395bf24fc	2018-05-25 14:11:54	PFCD-4345	PFCD-4345 ensure customerId is filled before the very first cwf run
3f8c2ed53fdee61	2018-05-23 16:33:50	PFCD-4345	PFCD-4345 set api.contextType in cw logic
09f451fc219468b	2018-05-17 11:08:59	PFCD-4345	PFCD-4345 Allow creation workflow logic to be run on save of Q/CT/RBA
c21fe3f71e216a4	2018-05-21 08:21:15	PFCD-4347	PFCD-4347 api.customerGroupEntry() / api.productGroupEntry cause Show definition not working
40a6e3fd1731da3	2018-05-23 12:14:08	PFCD-4351	PFCD-4351 INPUTMATRIX in Configurator doesn't show up properly.
e2b58719178e39b	2018-05-23 12:11:17	PFCD-4351	Revert "PFCD-4351 INPUTMATRIX in Configurator doesn't show up properly." This reverts commit 6840206

68402065e686c8e	2018-05-23 11:53:43	PFCD-4351	PFCD-4351 INPUTMATRIX in Configurator doesn't show up properly.
d29ef092c72e47e	2018-05-23 16:24:55	PFCD-4353	PFCD-4353 Make ChartBuilder's Heatmap more lenient if no SortBy definied
cdba9f0c02a79f3	2018-05-23 16:23:40	PFCD-4353	PFCD-4353 Add unit tests for ChartBuilder's Heatmap class
13b3557311ce7f1	2018-05-23 16:22:40	PFCD-4353	PFCD-4353 Fix ChartBuilder for Heatmap.getSeries()
04284cefe8ae470	2018-05-28 12:01:38	PFCD-4358	PFCD-4358 when you once fill in the value it is not possible to delete it
726c7da89d43beb	2018-06-18 15:51:39	PFCD-4360	Revert: PFCD-4360 (and others): Backporting latest EC to master
462906b33df57b1	2018-06-11 10:02:58	PFCD-4360	PFCD-4360 (and others): Backporting latest EC to master
dec170df5c78226	2018-05-29 14:13:52	PFCD-4368	PFCD-4368 Add support to download excel files via dashboard controller button
d19c2cb1da900c2	2018-05-30 06:16:15	PFCD-4369	PFCD-4369 do not instantiate the other object service if not needed
9e47cc007eb97b0	2018-05-29 11:12:45	PFCD-4369	PFCD-4369 attribute meta read of object from global partitions fails
627e90d0f9c70fb	2018-05-30 10:48:23	PFCD-4372	PFCD-4372 - Sonoco: Effective Date of Copied Quote incorrect Fix and IT
aef25deb13f550b	2018-06-04 12:13:16	PFCD-4386	PFCD-4386 Product Group picker does not work correctly within Configurator
34b38d5e36c568d	2018-06-15 15:01:06	PFCD-4388	PFCD-4388 avoid NPE in DimensionFilterSelectItem
762c9fcd276e6ab	2018-06-13 07:07:45	PFCD-4393	PFCD-4393 XLS Export contract from contract overview does not contain Customer/Product fields
dc6e924c593a368	2018-06-07 13:30:40	PFCD-4394	PFCD-4394 Configurator: Multitier fields are clickable even though quote is submitted/approved
b1b97effefd2bfe	2018-06-08 12:19:12	PFCD-4395	PFCD-4395 Impossible to select another preferences set when PL was based on template PL preferences
ec37f307f381273	2018-06-11 08:32:21	PFCD-4404	PFCD-4404 set alignment to right
fe84a04fa28ba73	2018-06-19 09:23:35	PFCD-4430	PFCD-4430 Embedded Dashboards: make sure parameters can be serialized If a response cannot be serialized, the server will reply with an empty response. Now throwing an exception at DashboardApi.setParam() level so the response contains an appropriate error message.
9450e9cf87bc14e	2018-06-19 09:19:24	PFCD-4430	PFCD-4430 DashboardService in classic UI: handle empty server response empty response my happen when server is unable to serialize the calculation result
f5dd0b454291461	2018-06-21 06:16:28	PFCD-4431	PFCD-4431 Make response parsing more robust
8314df0e798a062	2018-06-26 08:46:10	PFCD-4433	PFCD-4433 Change default behaviour
3fb189e87002174	2018-06-25 11:01:35	PFCD-4433	PFCD-4433 Multiple issues with InputType.OPTIONS
69505d5590fb8b8	2018-06-27 06:46:51	PFCD-4437	PFCD-4437 Filling in Product field manually via keyboard does not trigger onChange event and recalculation of the configurator logic
0f14c29c119f91c	2018-06-28 14:46:25	PFCD-4440	PFCD-4440 - Products filtering on the PX fields ignores the PX field type improved operator
38704cbb67ff22c	2018-06-28 14:46:18	PFCD-4440	PFCD-4440 - Products filtering on the PX fields ignores the PX field type Enabling date-picker for PX/CX date fields in Filter Builder
e6af63dc2f64937	2018-06-26 11:28:22	PFCD-4442	PFCD-4442 Weird sorting of OPTION input when labels are applied
08448834dc9bf0a	2018-07-31 15:59:09	PFCD-4450	PFCD-4450: Removed hiding of filter builder in case filter is modified via text entry

f6ec33d48127bd4	2018-08-03 09:35:57	PFCD-4459	PFCD-4459 Fix for commit a2965e5 "avoid recreating chart instance..." Was a big copy/paste mistake, each case statement was checking for the same instance type. Unless the chart was a Scatter, the ChartEditor was still re-creating the Chart instance every time. Now it works as advertised and spares us about 1/10th of a second per chart.
89bb1b0a3de7c40	2018-08-03 06:34:30	PFCD-4459	PFCD-4459 Chart UI: avoid multiple calls to ChartEditor.initialize() Calling initialize() is now only needed when creating a new empty chart. ChartEditor.deserializeChart() and ChartEditor.restoreFromRecord() will take care of it internally.
a2965e5b84f2148	2018-08-02 17:19:50	PFCD-4459	PFCD-4459 Chart UI: avoid recreating chart instance on refresh
20cc7977b8b65b0	2018-08-02 17:05:21	PFCD-4459	PFCD-4459 Dashboards perf: avoid creating unused portlet instances We were creating portlet instances just for the sake of comparing the class type with existing instances and most of the time keeping the existing one.
95b315dccca456d	2018-08-02 16:02:11	PFCD-4459	PFCD-4459 Charts UI: destroy() UI elements that will not be used anymore More work will be needed but about 80% of "leaked" instances are now destroyed
1c6f9ce83cfe111	2018-08-02 14:41:30	PFCD-4459	PFCD-4459 Add some timing logs and a specify a few Canvas IDs The Canvas IDs are very usefull to identify what UI elements fails to be Canvas.destroy()ed (i.e. garbage collected). The IDs the be spotted in either the SmartGWT Developer Console's Watch tab, or in TabIndexManager.getAllocatedTabChain()'s content.
9d09ca97e2d6d79	2018-08-02 08:33:46	PFCD-4459	PFCD-4459 Avoid sgwt warning AdvancedListGrid is a VLayout, so align to left or right makes no sense.
ade17ddb636a899	2018-08-02 08:22:07	PFCD-4459	PFCD-4459 Avoid NPE
17a226b1350a23d	2018-07-10 11:06:40	PFCD-4474	PFCD-4474 Fix regression with api.find for "PXAM" GLOBAL PX
a4108c82bedf468	2018-07-10 06:35:52	PFCD-4474	PFCD-4474 Fix regression with api.find for "PXAM" GLOBAL PX
5416686791958e0	2018-07-16 09:58:47	PFCD-4482	PFCD-4482 Label changes
8c7126eb25f8ce5	2018-07-12 09:01:25	PFCD-4482	PFCD-4482 Label change
096a4e267807f78	2018-07-12 06:24:18	PFCD-4485	PFCD-4485
f5b1c17ad91a3f8	2018-07-23 09:10:16	PFCD-4488	PFCD-4488 JSON output of String instead of real list when output is rawjson
939225be8a14f57	2018-08-09 14:42:36	PFCD-4489	PFCD-4489 - Wrong translation files - incorrectly translated key values
c549cd21b33e27e	2018-07-23 15:50:36	PFCD-4489	PFCD-4489 - Wrong translation files - incorrectly translated key values (Spanish)
a073eecb75c0507	2018-07-23 11:03:41	PFCD-4491	PFCD-4491 Expose active calc results of (X)PGI to groovy
a3d02c804082ba6	2018-07-30 07:54:00	PFCD-4493	PFCD-4493 In case of values restricted by filter the input format type will be still working
a1295d46dafdd55	2018-07-17 09:56:05	PFCD-4500	PFCD-4500 User Group Override is not working in UI
0e29988aac44b05	2018-08-10 08:21:03	PFCD-4505	PFCD-4505 Backport from develop PFCD-4558 Add logging to help issue resolution
b39d986f625306a	2018-07-20 09:31:32	PFCD-4507	PFCD-4507 Quote Header Text is cached from other quote
3d05a582d9329af	2018-07-20 06:59:04	PFCD-4509	PFCD-4509 Set vertical autofit to actual records
d6cfdd72378f321	2018-07-30 11:18:25	PFCD-4522	PFCD-4522 Not possible to Download Excel when filtering on date field with null, not null, between and between (inclusive)
ea1abec6acc4630	2018-07-30 13:24:00	PFCD-4534	PFCD-4534 (DE) Translation of new Unity labels
00904eba1c8943a	2018-07-27 05:57:24	PFCD-4535	PFCD-4535 SSO Login when client session exists may lead to uninitialized UI
0c6a7abf2875d23	2018-07-31 07:01:22	PFCD-4536	PFCD-4536 forgotten hunk

ecdf787874f72c5	2018-07-31 07:01:14	PFCD-4536	PFCD-4536 allow creation workflow to be generated after all the previous runs generated empty creation workflows
b8b56c3faf820be	2018-07-30 12:27:31	PFCD-4537	PFCD-4537 header outputs should not be cleared on fetch
54990a8ad221768	2018-08-01 11:30:34	PFCD-4544	PFCD-4544 User groups assigned by Business Roles are not applied in the workflow
2d1240109ffab43	2018-08-17 12:54:15	PFCD-4546	PFCD-4546 Do not show Invalidated records in My Pending Approvals
988dbcdbc68de35	2018-08-03 11:08:40	PFCD-4546	PFCD-4546 Fixed also for Contract and Rebate Agreement
352815714172ebb	2018-08-03 10:57:40	PFCD-4546	PFCD-4546 WF record status not changed to Invalidated one quote is is Invalidated
3235c3caa04a35a	2018-10-11 06:57:22	PFCD-4547	PFCD-4547 Workflow: Quote approved via email link caused Server Error 500
473967521839432	2018-08-09 08:58:56	PFCD-4553	PFCD-4553 ChartBuilder API: Fixed the generated Heatmap custom colors ...and the matching Unit test.
94b90302bed7e0d	2018-08-30 07:34:30	PFCD-4557	PFCD-4557 Gauge is not rendered inline in a Quote Tree View items overview
0e29988aac44b05	2018-08-10 08:21:03	PFCD-4558	PFCD-4505 Backport from develop PFCD-4558 Add logging to help issue resolution
f8222ddc2551ca0	2018-08-18 09:34:13	PFCD-4560	PFCD-4560 (backporting) allow api.httpCall additional headers
f2230ffd3e56dfd	2018-08-30 07:38:58	PFCD-4565	PFCD-4565 Allow Gauge column to be added to Quote line items Table view
54ad823a0d91925	2018-08-14 13:45:22	PFCD-4572	PFCD-4572 viewlog API endpoint has no permission restriction
14877b7d33df232	2018-08-16 14:33:51	PFCD-4579	PFCD-4579 Add FilterOperator.NOT_IN_SET
9cc06cfab2118a4	2018-08-16 14:20:33	PFCD-4580	PFCD-4580 Can't create new charts through the PriceAnalyzer menu anymore
a1d19c9e4bc29d3	2018-08-22 08:32:13	PFCD-4584	PFCD-4584 Avoid NPE when embedded popup dashboard was removed already
f7629fead25dfe1	2018-08-22 12:55:23	PFCD-4585	PFCD-4585 fix non-progressing cw
c905b9c434b064f	2018-08-21 07:40:16	PFCD-4585	PFCD-4585 creation workflow, allow reset and finish on empty
c367d30a9554418	2018-08-21 12:18:36	PFCD-4586	PFCD-4586 Avoid NPE in WaterfallChart.removeDictionaryEntries()
badf3303f5997d7	2018-08-30 10:55:36	PFCD-4600	PFCD-4600 Max dashboard count in dropdown in PA 75->300
0417613017a9b9c	2018-08-31 12:11:49	PFCD-4604	PFCD-4604 conditionally show progress during configurator recalculation
5dda91c36b318f9	2018-08-31 11:49:59	PFCD-4604	PFCD-4604 conditionally show progress during configurator recalculation
a58639053d0c084	2018-08-31 11:36:46	PFCD-4604	PFCD-4604 conditionally show progress during configurator recalculation
ca05bad6b40f798	2018-09-09 18:48:02	PFCD-4614	PFCD-4614: PA: gracefully handle stale cache entry removal DB error in calculation slave thread
3579bd873609783	2018-04-20 10:44:23	PFSFDCQC-1053	feat(unity): add l10n (PFSFDCQC-1053)
3d1fdf7680aac82	2018-06-05 11:20:10	PFSFDCQC-1294	PFSFDCQC-1294 translations
0b04ab0fbcd4043	2018-07-11 12:30:22	PFSFDCQC-1315	PFSFDCQC-1315 translations
3136e1359b7b242	2018-07-25 12:37:22	PFSFDCQC-1343	PFSFDCQC-1343: translations

544f7998cac09c5	2018-06-21 12:26:45	PFSFDCQC-1409	feat(unity): add l10n (PFSFDCQC-1409)
4ffcc6bff37b0b1	2018-07-10 10:53:51	PFSFDCQC-1474	fix(csrf) ping request is not checked with csrf (#PFSFDCQC-1474)
928e64ef1f8a71b	2018-07-23 13:51:10	PFSFDCQC-1502	fix(translations) correcting key and text for Pending Approvals #PFSFDCQC-1502
b8e52d73e63d432	2018-04-25 07:56:43	PFSFDCQC-622	PFSFDCQC-622 - modify core localizations; Change titles on filter icons

Cosmopolitan 3.3 Changelog

Commit	Date	Issue	Message
307840043b9c7a2	2017-12-22 13:47:07	PFCD-3285	PFCD-3285 Also migrate sku/customerId in business key migration
a363a4821a3f485	2017-11-14 14:46:43	PFCD-3439	PFCD-3439 Code cleaning
65bfbd5efcade25	2017-11-14 14:34:15	PFCD-3439	PFCD-3439 BubbleChart: show the "real values" instead of normalized ones The bubble size in bubble charts is normalized (i.e. values between 0 and 1). Now showing the real values in the tooltip and the data tab, and normalized value is hidden and used only to build the chart.
8035e2473f6ffa0	2017-11-14 14:26:31	PFCD-3439	PFCD-3439 PA: normalized values sometimes returned as -0 or Infinity Making sure -0 is "rounded" to 0.0 now, and zero divide (when all items have the exact same value) now results in 0.5 instead of Infinity.
497c330aa64fe9b	2017-11-30 17:39:41	PFCD-3440	PFCD-3440 additional measures support for Histogram's ChartBuilder hint ... and also a few additional chart options like ShowZAxis, ZAxisLabel, and Y and Z axis min-max values.
57a5ad33fbb47b1	2017-12-21 10:08:45	PFCD-3514	PFCD-3514 Add BarLineChart.setDisableDrilldown() to ChartBuilder API
5edeaf611d6cb14	2017-12-20 17:58:59	PFCD-3514	PFCD-3514 dashboard click event support for BarLine chart
4209217dd444126	2017-12-19 16:10:31	PFCD-3514	PFCD-3514 typo in SharedChartRepresentations
238528ee2aaa285	2017-12-19 16:10:30	PFCD-3514	PFCD-3514 initial DataTableChart.addEvent() row click support Allows access to the groupBy dimensions, one by one, via .withDimensionAsEventDataAttr(dimensionName)
2f8042930601a7a	2017-12-19 16:10:29	PFCD-3514	PFCD-3514 Add DataTableChart.setDisableDrilldown() to ChartBuilder API We will want that as showing the drilldown menu messes up the user experience when the click triggers a dashboard refresh
43598ff6da3a1ce	2017-12-19 16:10:28	PFCD-3514	PFCD-3514 small fixes/cleaning of ScatterChart click events code
e47c575b0232c4a	2017-12-12 16:23:18	PFCD-3534	PFCD-3937 Revert commit from PFCD-3534 "...fix data table ordering" Revert "PFCD-3534 simplify waterfall query and fix data table ordering" This reverts commit bca01ce3d85252d9811a83b15884ad98f39584c7.
496743a59ab564f	2017-11-23 17:12:17	PFCD-3638	PFCD-3638 ChartBuilder API TrendLine tests for bubble, scatter and time
8b330139db6a9a4	2017-11-23 17:09:26	PFCD-3638	PFCD-3638 ChartBuilder API support for BubbleChart TrendLine
eeb2cf2e11d6439	2017-11-23 14:45:19	PFCD-3638	PFCD-3638 ChartBuilder API fix TimeSeries TrendLine.setRegressionType()
cde234dd65e56be	2017-12-01 14:08:19	PFCD-3672	PFCD-3672 API doc updated
e6a723a0a932f4b	2017-12-05 15:09:51	PFCD-3822	PFCD-3822 "unhide" filter operators like isNull, notNull, between... Some SmartGWT filter operators like isNull are now hidden from the UI by default. This is a hack to make them visible again.
83d24480f67ac6f	2018-01-08 13:33:06	PFCD-3828	PFCD-3828 Fix recalculation request handling
da729fec24e3d38	2017-12-12 11:39:22	PFCD-3828	PFCD-3828 Extend "LPG Editor +" role to be able to modify PG configuration (formula inputs only!)
fc08587b08abd11	2018-01-17 06:29:23	PFCD-3840	PFCD-3840 Fix debug setting
7b03a1230d55068	2017-11-30 09:47:28	PFCD-3864	PFCD-3864 various Waterfall fixes and improvements - ensure measures ordering does not break because of invalid lines - avoid NPE in getCellCSSText() - do not query the server if measures grid has validation errors - add an error tooltip on rows with server validation errors

af0ca6c0fb70eeb	2017-11-30 09:47:27	PFCD-3864	PFCD-3864 fix two NPEs in ItemSpecifierController was sometimes blocking Waterfall chart editing
a28b509f1e85e3b	2017-11-30 09:47:26	PFCD-3864	PFCD-3864 fix/improve cell selection log messages in AdvancedListGrid
cc9ad2564fe27cd	2018-01-15 09:53:18	PFCD-3872	PFCD-3980 Excel Client - filtering for DCR including backports for PFCD-3935, PFCD-3949, PFCD-3905, PFCD-3872
6b4644e648c6f7f	2017-12-08 14:31:32	PFCD-3890	fix(RebateRecordsPageViewImpl): RR View/Edit details: Arrow icons don't work (PFCD-3890)
68222186ef60536	2017-12-04 09:44:51	PFCD-3898	PFCD-3898-profiler-add-call-count
cc9ad2564fe27cd	2018-01-15 09:53:18	PFCD-3905	PFCD-3980 Excel Client - filtering for DCR including backports for PFCD-3935, PFCD-3949, PFCD-3905, PFCD-3872
c935d3b5b6568a5	2017-12-05 09:59:29	PFCD-3907	PFCD-3907 PA field with expression (ie not persisted) can be a dimension
48c5ffc1c75fba6	2017-12-06 12:24:12	PFCD-3914	fix(PFCD-3914): Day calendat field not resolved correclty in DM view
22fb5c154a7799a	2017-12-07 12:23:19	PFCD-3915	PFCD-3915 do not fail when constructing client representation with deleted users/groups
c3288223750d797	2017-12-07 10:03:44	PFCD-3915	PFCD-3915 do not allow deleted user to login on wf direct action
afe06e0af276678	2017-12-06 21:26:15	PFCD-3915	PFCD-3915 Deleted user which was not deactivated is able to approve workflow via direct link
46ee5f7e032ec2b	2018-03-01 12:17:27	PFCD-3920	fix(PFCD-3920) : PA-calculation-errors-leads-to-DB-connection leak
320de4129ced076	2017-12-06 21:09:19	PFCD-3922	PFCD-3922 wf with no reason set and watcher step fails on npe
5ca06b95d3f4258	2017-12-18 12:28:56	PFCD-3931	PFCD-3931 correcting merge issue
bceee300e294078	2017-12-18 12:06:12	PFCD-3931	PFCD-3931 reinstating unintentionally dropped chunk
ca08a6650a4a31e	2017-12-18 11:25:29	PFCD-3931	PFCD-3931 reverting a few chunks in previous commit as incorrectly merged and not relevant to the fix
a1149ca9d88cf70	2017-12-18 10:48:54	PFCD-3931	PFCD-3931 fixed
8615cc73403fb38	2017-12-18 10:47:20	PFCD-3931	PFCD-3931 fixed
2047d88027ccb9f	2017-12-13 23:48:49	PFCD-3931	PFCD-3931: chart built with chart builder API results in invalid X or Y when it's an expression. Fixed for X/Y (i.e. ChartFields) but not yet for AdditionalMeasures.
854adcbd7b300e7	2017-12-21 09:55:24	PFCD-3934	PFCD-3934 Few "Custom Calculated Fields" enhancements
cc9ad2564fe27cd	2018-01-15 09:53:18	PFCD-3935	PFCD-3980 Excel Client - filtering for DCR including backports for PFCD-3935, PFCD-3949, PFCD-3905, PFCD-3872
5cca7e667a04729	2017-12-12 16:33:12	PFCD-3937	PFCD-3937 comment explaining why we had to revert WaterfallQuery changes
00acf2dfe708737	2017-12-12 16:23:19	PFCD-3937	PFCD-3937 Align ordering of Waterfall data between chart and grid... while also avoiding "Illegal Operator" errors at query time.
e47c575b0232c4a	2017-12-12 16:23:18	PFCD-3937	PFCD-3937 Revert commit from PFCD-3534 "...fix data table ordering" Revert "PFCD-3534 simplify waterfall query and fix data table ordering" This reverts commit bca01ce3d85252d9811a83b15884ad98f39584c7.
84a0cf91925458d	2017-12-13 09:47:13	PFCD-3939	PFCD-3939 sort change leads to scroll left in pgi grid
a9a38794e77a228	2017-12-14 11:39:27	PFCD-3944	PFCD-3944 poststep logics names confgiurable instead of hardcoded
ee6fb284da846e4	2017-12-13 15:43:34	PFCD-3944	PFCD-3944 no json conversion for perf reasons
a52d5c35d0c596c	2017-12-13 15:05:45	PFCD-3944	PFCD-3944 post step login on empty workflow and on no workflow

cc9ad2564fe27cd	2018-01-15 09:53:18	PFCD-3949	PFCD-3980 Excel Client - filtering for DCR including backports for PFCD-3935, PFCD-3949, PFCD-3905, PFCD-3872
83d95c2e1632b15	2018-01-05 10:12:52	PFCD-3958	fix(DownloadRequestFactory): Cannot select a tab in Pricefx after downloading content of a table to Excel (PFCD-3958)
981ed0327a3583c	2017-12-20 11:35:01	PFCD-3959	PFCD-3959 workflow fails if workflow delagation is not used
e8f0100826bad0d	2017-12-21 10:22:43	PFCD-3961	PFCD-3961 api.getSecondaryKey() always NULL in formula test drive
0f8b6b05a9a417c	2017-12-21 14:28:31	PFCD-3962	PFCD-3962 Grid: left-scrolling upon a column size change when 2-3 column frozen
d7faee92cc5bfcd	2017-12-22 13:40:09	PFCD-3964	PFCD-3964 optimize
f777446617b63dd	2017-12-22 12:34:13	PFCD-3964	PFCD-3964 a commit must happen after dcr apply and before event item_approved emit
cc9ad2564fe27cd	2018-01-15 09:53:18	PFCD-3980	PFCD-3980 Excel Client - filtering for DCR including backports for PFCD-3935, PFCD-3949, PFCD-3905, PFCD-3872
da0cfb8e881e459	2018-01-15 15:40:00	PFCD-3983	fix(WorkflowUtil): Type and link to PGI is not visible in My Pending Approvals for Matrix LPG (PFCD-3983) Added support for XPGI (original code only handled PGI).
645754f13ceb700	2018-01-15 13:30:22	PFCD-3988	PFCD-3988 ignore IT on postgres for now
0bdd11ebda038f6	2018-01-15 13:16:17	PFCD-3988	PFCD-3988 mltv find with aggregation with meta fix
6bacbf96a8fc725	2018-01-15 13:20:47	PFCD-3990	PFCD-3990 set current user as executedByUser into templating engine
409f8bb961624e6	2018-01-15 13:19:35	PFCD-3990	PFCD-3990 ensure that PGI/XPGI has always createdBy filled
e4d6f43af3c9381	2018-01-15 12:44:47	PFCD-3992	PFCD-3992 EC: User cannot type DCR name
bbc8621794cbac5	2018-01-15 15:45:34	PFCD-3994	PFCD-3994 fix flexcharts template loading with proper try/catch block
89faf4e67cd36d4	2018-01-16 12:50:37	PFCD-3997	PFCD-3997 "Whitelist" fields "userGroupEdit" and "userGroupViewDetails" for updates with quoteProcessor.updateField() function
3114aaaf0cd197a	2018-01-24 08:06:52	PFCD-4007	PFCD-4007 Missing change
b33610fcdc8510a	2018-01-22 12:09:13	PFCD-4007	PFCD-4007 Show and apply prefs also on Matrix portlets on dashboards
e0bcd9ec2d9a85b	2018-02-06 15:22:24	PFCD-4009	PFCD-4009 DimensionFilterSelectItem fix refresh button It had become unresponsive due to caching after the previous changes in this branch.
e10cad758d61766	2018-02-01 13:29:52	PFCD-4009	PFCD-4009 remove some logging
1772a3bf6c951e0	2018-02-01 08:57:35	PFCD-4009	PFCD-4009 Improvements to DimensionFilterSelectItem (like validation) DimensionFilterSelectItem does not extend DimensionValueSelectItem anymore. It doesn't reset its value on refresh anymore, but instead it validates its value.
795921dd9ddd52d	2018-01-23 13:23:36	PFCD-4009	PFCD-4009 warning: left align of VLayout makes no sense not related to the PFCD issue, but still in the charts
ecec0af8ee11a8c	2018-01-22 11:20:47	PFCD-4009	PFCD-4009 get rid of a loosely related Record vs ListGridRecord warning
3edbfdd5f6eb423	2018-01-19 17:21:11	PFCD-4009	PFCD-4009 picklist filter was lost on DimFilterSelectItem refresh
0b4400f06f0be58	2018-01-22 20:37:25	PFCD-4010	Revert "PFCD-4010 one more codepath" This reverts commit 9f5e19fef22c85e2662a9f53b95a370a3a9b6cf4.
dd6a06d9feb3785	2018-01-22 20:37:12	PFCD-4010	Revert "PFCD-4010 additional logging" This reverts commit a76afdd70792f4e42328605bcef1feeef62d0e7f.
6ff876d19d3f937	2018-01-22 20:36:55	PFCD-4010	Revert "PFCD-4010 typo" This reverts commit 182d152bd8a578030ecf6ea4a230ecc0be9f7512.
182d152bd8a5780	2018-01-22 11:11:54	PFCD-4010	PFCD-4010 typo

a76afdd70792f4e	2018-01-22 11:11:04	PFCD-4010	PFCD-4010 additional logging
9f5e19fef22c85e	2018-01-22 10:47:24	PFCD-4010	PFCD-4010 one more codepath
4da8dc9186afcfb	2018-01-21 21:01:03	PFCD-4010	PFCD-4010 Allow Workflow Post Step Logic Formula processing also without obligation to define WF - for autoapproved
8af772fd1afe6cd	2018-01-23 07:13:52	PFCD-4013	PFCD-4013 Always set runNumber to 1 when starting a dirty pass
87799a708daed00	2018-01-23 13:19:17	PFCD-4016	PFCD-4016 Error in console while opening LPG Overview page
6dc3dbc5cc2b0db	2018-01-23 21:38:47	PFCD-4017	fix(PFCD-4017) : PA Histogram chart - cumulative measure not shown correctl
5e1dfcb01ac7012	2018-02-05 14:25:49	PFCD-4018	PFCD-4018 GroupByItem: simplify
593b69e8d640bcc	2018-02-05 11:35:46	PFCD-4018	PFCD-4018 GroupByItem: less logging and some cosmetic code changes
c624eb9a28b44c4	2018-02-05 11:28:49	PFCD-4018	PFCD-4018 PA Data Table: enabling/disabling of Add Group-By button - enable/disable the button instead of hiding it, which allows to display a tooltip message explaining why it's disabled - fixed remaining issues with the enable/disable checks of the button
8af2e39aa192e4f	2018-02-01 18:24:21	PFCD-4018	PFCD-4018 PA Data Table: fix dictionary updates when editing group-bys The fix required to change GroupByItem's notifications to be triggered after the changes are visible for getGroupByNames() instead of just before. That's probably more intuitive anyway.
43e192b630ec4f2	2018-02-01 16:49:17	PFCD-4018	PFCD-4018 PA Data Table: enforce max group-by limit (disable add button) + fixes a notification issue when editing a group-by field + some cosmetic code changes
740b0b605193a2b	2018-01-24 12:47:02	PFCD-4021	PFCD-4021 Allow AppProperty values to be returned "raw"
caaa49c59765f06	2018-03-26 15:10:12	PFCD-4024	fix regression in PFCD-4024
0a5bd180fcadaf4	2018-02-22 17:12:47	PFCD-4024	PFCD-4024 Comparison Waterfall legend was missing Also now removed the Hide Legend option from the simple Waterfall chart, as we never show a legend there anyway.
4b9816d1819a939	2018-02-23 06:46:24	PFCD-4026	PFCD-4026 Do not perform dirty item recalculation of currently updated item immediately if there are other dirty items as well
1218c4a89db9781	2018-02-01 20:52:39	PFCD-4027	fix(PFCD-4027)
0fe190829d80dd8	2018-01-31 07:45:28	PFCD-4033	PFCD-4033 PriceRecord loses customerId upon reconversion
e3f1208fdce78c5	2018-02-01 13:03:21	PFCD-4037	fix(PFCD-4037)
2bc59adab46a5f1	2018-02-02 13:37:24	PFCD-4039	fix(web): QC - Adding products containing "periods" via text entry is not possible (PFCD-4039)
588878efa010643	2018-02-12 09:21:26	PFCD-4056	PFCD-4056 Set dirty flag lock-safe
17b499e4664dccf	2018-02-14 15:15:05	PFCD-4065	PFCD-4065 InputType.CUSTOMER loses focus while typing inside a configurator
34a45592dfc6a99	2018-03-07 16:02:39	PFCD-4070	PFCD-4070 missing file
7e9e70a3a9ea987	2018-03-07 15:54:19	PFCD-4070	PFCD-4070 autogroup capability for calc results details popup
4d3b30229dd85b2	2018-02-21 12:22:50	PFCD-4078	PFCD-4078 forgotten hunk
1b12e1927c1d0cf	2018-02-21 12:20:15	PFCD-4078	PFCD-4078 adjust oracle and pg
95c5ac3cb64d1a8	2018-02-21 10:24:58	PFCD-4078	PFCD-4078 mariadb safe equal does not break index usage
4005f129110a6f3	2018-02-19 10:56:19	PFCD-4086	PFCD-4086 Close groovy sandbox dangerous code paths

a426f0dcda8ca4c	2018-03-15 22:25:29	PFCD-4090	fix(PFCD-4090): PA calculationL MVStore StackOverflow errro
d68191686af504f	2018-02-22 20:32:05	PFCD-4100	PFCD-4100 WF Delegation - email might not be sent on activation
d9258d33dbb742b	2018-03-18 14:59:20	PFCD-4113	fix(PFCD-4113) : PA calculation: do not time out feeder formula (did not work outside of the IT)
2e86ea1a9965458	2018-03-01 13:51:14	PFCD-4113	fix(PFCD-4113): PA calculation: do not time out feeder formula
46e383525a0a2dd	2018-02-28 16:49:51	PFCD-4118	PFCD-4118 Allow "mark clean" also for other items (similar to "mark dirty") - Cherry pick
c1029ad4f97ea88	2018-02-28 16:42:33	PFCD-4119	PFCD-4119 Set item "clean" also on calculation failure (per-instance configurable)
d019b69e596d222	2018-03-21 11:57:59	PFCD-4124	PFCD-4124 PX50s not visible in dcr type picker
7e66cace406e408	2018-03-19 11:09:38	PFCD-4124	PFCD-4124 support first 30 attributes of PX50/CX50 in DCR
cbfad08977a56f4	2018-03-19 17:11:05	PFCD-4134	PFCD-4152 PFCD-4134 restore ListGridItem's default height
da36b7978235dcc	2018-03-08 08:25:45	PFCD-4134	PFCD-4134 InputMatrix make it auto-adjust height based on pre-seeded values and display options set
5bbef6a6104dfb0	2018-03-07 15:52:44	PFCD-4134	PFCD-4134 Enahnce InputMatrix
eef54719db75b62	2018-03-13 08:14:13	PFCD-4147	PFCD-4147 Failover for CalculationFlow processor PFCD-4147 Failover for CalculationFlow processor - the adapter PFCD-4147 fixes PFCD-4147 info log level for normal processing on master
cbfad08977a56f4	2018-03-19 17:11:05	PFCD-4152	PFCD-4152 PFCD-4134 restore ListGridItem's default height
424c46510b34131	2018-03-13 15:33:34	PFCD-4155	PFCD-4155 inc file conversion service timeout 2x
83c4c6e77861ef4	2018-03-13 12:02:59	PFCD-4155	PFCD-4155 file conversion service configurable timeouts
729dc95352609c7	2018-03-14 12:37:08	PFCD-4157	PFCD-4157 InputMatrix option: noCellRefresh to prevent configurator round trips per cell edit
efa4fbae874db87	2018-03-16 10:51:54	PFCD-4160	PFCD-4160 product filter block quote save when special chars are present in skus
4cb0f1d034335f7	2018-03-16 10:51:06	PFCD-4163	PFCD-4163 avoid npe on quote fetch product reported when cg inut with no value present on header
6ee072b720157c3	2018-03-18 14:24:05	PFCD-4166	fix(PFCD-4166) : PA: streamQuery needs intermediate commits when storing data in a file backed map
682c81be69ecda2	2018-03-19 19:47:14	PFCD-4170	fix(FormulaManagerService): Translation Editor: Language specific labels not copied when copying a CL using copy button (PFCD-4170) Simple fix and added also to IT
efe5e96a764ed65	2018-04-12 07:29:27	PFCD-4240	PFCD-4240 Caught LinkageError on parsedscript_... problem ... v.2
5977e98f628a5e5	2018-04-17 13:05:00	PFCD-4259	PFCD-4259 Strict CSRF check breaks XLS download (stupid typo)
c3ddf21d06294b7	2018-04-17 12:56:09	PFCD-4259	PFCD-4259 Strict CSRF check breaks XLS download
536754ffcdcf15e	2018-04-18 15:19:57	PFCD-4263	PFCD-4263 Formula fixed inputs missing in chart's calculation widget
270861b77bc31f5	2017-12-15 13:27:04	PFSFDCQC-448	PFSFDCQC-448 Clear local storage on logout in classic UI
4b718bd96d5ef1d	2017-12-15 13:26:51	PFSFDCQC-448	PFSFDCQC-448 store CSRF also in local storage of classic UI
51a6287fc4bdf2e	2018-03-14 11:32:06	PFSFDCQC-690	feat(unity): added i18n (PFSFDCQC-690)
1dfccd2f64430cf	2018-03-19 11:38:31	PFSFDCQC-967	feat(unity): i18n (PFSFDCQC-967)

Mojito 3.2 SE Changelog

Commit	Date	Issue	Message	
a4382af83b90be8	2017-11-28 14:06:57	ORA-01795	PFCD-3868 ORA-01795 more than 1000 values in IN clause	
118bb607a5760b7	2017-10-19 15:35:55	ORA-01795	PFCD-3745 - User can not log in when he is member of 1000+ groups (ORA-01795) Removed wrong import!	
16438db898c4581	2017-10-19 15:27:45	ORA-01795	PFCD-3745 - User can not log in when he is member of 1000+ groups (ORA-01795) Hot fix + improved IT to cover both cases (number of groups below 1k and over 1k)	
86c6711f112e197	2017-10-19 15:27:38	ORA-01795	PFCD-3745 - User can not log in when he is member of 1000+ groups (ORA-01795) Created IT for to reproduce this on Oracle.	
4ac992ae5593d03	2017-10-10 15:42:57	PCFD-3691	PCFD-3691 forgotten hunk	
c30ea96f145e0b1	2017-10-13 11:18:12	PFCD-2398	fix(PFCD-2398)	
8cf55798b1e84a8	2017-10-05 11:13:49	PFCD-2398	PFCD-2398 : filter dropdown query created in backend did not include sort clasue	
3f811a8d0ada9c6	2017-10-12 09:46:48	PFCD-3104	PFCD-3104: policy record creation from model now AttribMeta aware	
a0881fd8ad3f812	2017-10-10 09:02:44	PFCD-3104	PFCD-3104: PO missing Role i18n	
57d24433962d015	2017-10-09 15:36:31	PFCD-3104	PFCD-3104: PO row level analysis fix	
11e20de4d7bb344	2017-10-05 10:39:04	PFCD-3104	PFCD-3104: segment tree creation fix	
92b28193822359c	2017-09-28 13:31:13	PFCD-3104	PFCD-3104: permissionm fix for PO MGR role	
256e7b6512e2ffa	2017-09-28 09:55:26	PFCD-3104	PFCD-3104: fix grid prefs in RollupQueryEditor	
8db95cbf231bf81	2017-09-20 07:51:08	PFCD-3104	PFCD-3104 fix integration test	
afc1931f4298638	2017-09-19 12:15:22	PFCD-3104	PFCD-3104: fix in RollupQueryEditor	
8a5a2b572883513	2017-09-18 10:11:57	PFCD-3104	PFCD-3104: minor mods following PR-116 feedback	
b7500a3aa8c47b0	2017-09-18 09:11:47	PFCD-3104	PFCD-3104: fixes	
cf0cf85c071c714	2017-09-17 12:04:56	PFCD-3104	PFCD-3104: DB migration script for DMRollup additional fields	
b495d29d1253dc3	2017-09-17 10:37:57	PFCD-3104	PFCD-3104: fixes	
15e01a404e4471c	2017-09-15 12:32:32	PFCD-3104	PFCD-3104: fixes	
0c58a799bbe8d8b	2017-09-15 10:11:24	PFCD-3104	PFCD-3104: tweaks and fixes	
f91a7756f1cd77e	2017-09-14 21:38:35	PFCD-3104	PFCD-3104: tweaks and fixes	
a6bbc5a14b3728e	2017-09-14 20:15:36	PFCD-3104	PFCD-3104: tweaks and fixes	
fe8fd5b4a3935a0	2017-09-14 18:48:37	PFCD-3104	PFCD-3104: tweaks and fixes	
1a58ecfddf3e77d	2017-09-07 15:13:33	PFCD-3104	PFCD-3104: UI fix (ModelRecordMeta) - missing file in previous commit	
fbbe5b39890e81c	2017-09-07 09:24:33	PFCD-3104	PFCD-3104: UI fix (ModelRecordMeta)	

dcc620b52e338d0	2017-09-06 07:26:36	PFCD-3104	PFCD-3104: UI fix (Summary view)	
23865e56465a106	2017-09-05 21:42:02	PFCD-3104	PFCD-3104: UI fix (Summary view)	
bc594c66fd7801d	2017-09-05 20:23:05	PFCD-3104	PFCD-3104: UI fix (Summary view)	
7faf418a9171603	2017-09-05 17:04:17	PFCD-3104	PFCD-3104: couple of additional helper methods in Matrix2DPFCD-3104: couple of additional helper methods in Matrix2D	
2fc545658e5b7b5	2017-09-05 17:03:23	PFCD-3104	PFCD-3104: couple of additional helper methods in Matrix2D	
3aa1d6a4b72e8de	2017-09-05 17:02:24	PFCD-3104	PFCD-3104: couple of additional helper methods in Matrix2D	
3235a4edf0a6bd9	2017-09-04 12:40:51	PFCD-3104	PFCD-3104: fixes	
c484b30af122f96	2017-09-03 22:25:14	PFCD-3104	PFCD-3104: fixes	
69e7b4e95df0573	2017-09-26 15:34:42	PFCD-3248	PFCD-3248 charts with multiple dim filters show 'outdated' options	
aaa6b222df68eda	2017-09-03 22:26:30	PFCD-3263	PFCD-3263: fix (was not filtering JSTs initially)	
6e627e2c3755ecd	2017-09-12 11:50:42	PFCD-3264	PFCD-3264: regression fix: filter value was lost when widget was rendered multiple times	
5aaad866b659741	2017-09-12 09:14:55	PFCD-3264	PFCD-3264: fix: base (generic) filter was ignored in dim-filter dropdowns	
2280b5c9e6cdfc3	2017-09-15 07:32:41	PFCD-3319	PFCD-3319 Allowing client-only sort in Customer Assignment Editor now	
d199145a38c6eaf	2017-10-04 15:20:40	PFCD-3347	PFCD-3347: minor UI fix: marginally increase section width in PA Rollups UI	
e3c92de6a442702	2017-10-04 14:35:14	PFCD-3347	PFCD-3347: grid prefs reset with each query refresh (view button) + flagging unsupported queries server side	
219cecfdb554a26	2017-09-27 18:15:38	PFCD-3347	PFCD-3347 fix: dim filter values were not added to overall filter for Rollup	
d70c0629797a71d	2017-09-25 13:01:08	PFCD-3347	PFCD-3347: minor fix: do not allow section collapse in Rollups UI	
36274ce8af5cf2b	2017-09-25 12:51:20	PFCD-3347	PFCD-3347: minor fix: do not show "Draft" field in rollup grid	
515e7237f455176	2017-08-24 15:25:24	PFCD-3347	PFCD-3347 fixes	
d6e7aa1ed0b67ba	2017-09-27 18:12:59	PFCD-3364	PFCD-3364: downstream dim filter values were not refreshed when the upstream filter's value is cleared.	
292d08168f0d4b6	2017-08-28 09:15:29	PFCD-3364	PFCD-3364: regression fix	
14e9dd4f70eb908	2017-09-06 14:24:06	PFCD-3366	PFCD-3366 Fix histogram chart's additional measures tooltips	
bc082bc9de3cad2	2017-09-26 13:28:17	PFCD-3378	PFCD-3378 Prevent mass-edit approval for PGI that have resultPrice == NULL (and make it working for XPGI as well)	
f9d649179e46c7b	2017-08-25 14:02:46	PFCD-3458	fix(CleanStaleDataTask): Even though Cleanup deletes data ZIP file is blank (PFCD-3458) Problem was that the copy of the ZIP file meant to be uploaded to AWS to be shared in cluster was copied before the ZIP file was properly closed, hence empty copy of ZIP was created and that was returned by browser.	
89e653e717f9dd6	2017-09-01 08:20:40	PFCD-3461	PFCD-3461 fix npe	
7bcf001b751b164	2017-08-31 12:15:45	PFCD-3461	PFCD-3461 simple filter fails doing local filtering and using and extension fields	
a8ae502414e69f1	2017-08-25 11:47:06	PFCD-3471	PFCD-3471 fix removal	

6c47a313af3aef1	2017-08-29 12:15:00	PFCD-3494	PFCD-3494 sort on numerical attributes in PX/CX does not work	
6178cdd1b8d99fb	2017-09-20 08:42:37	PFCD-3500	PFCD-3500 Ability to disable triggering OBJREF generation on PR save	
b4c01002addb5b1	2017-09-05 07:26:56	PFCD-3506	PFCD-3506 detect join fields, allow part of key to be defined also in business key	
1fd5bfb6582d8b8	2017-09-01 13:55:39	PFCD-3506	PFCD-3506: Added new property to LoaddataOptions in swagger api schema	
8e9103efe8ae74c	2017-08-28 09:25:45	PFCD-3513	PFCD-3513 Show all groups in "view groups" selector also in overview Fixed regression introduce by previous commit (=not possible to edit User groups from Overview page at all)	
5b37b4f34750327	2017-08-25 10:10:28	PFCD-3513	PFCD-3513 Show all groups in "view groups" selector also in overview	
62079e7d1d1d060	2017-08-30 15:22:56	PFCD-3533	fix(i18n): German locale Mass Edit information message is misleading users to use "." as a decimal separator (PFCD-3533)	
40ef647200d6f79	2017-08-30 15:03:58	PFCD-3535	fix(SimpleUser): Rebate Calculation Logic menu item not visible although role Rebate Calc Logic Admin assigned (PFCD-3535)	
ffe78635ae09a43	2017-08-30 15:16:31	PFCD-3536	fix(role Promotion Manager): Promotion Manager role does not contain PRICINGFORMULA_FETCH (PFCD-3536)	
a7e14a4709827a2	2017-09-20 07:05:58	PFCD-3537	PFCD-3537 allow deselection of both C and CG input adding	
56175c1b747ecdf	2017-09-18 14:25:37	PFCD-3537	PFCD-3537 no unequal customerIdd values from inputs allowed	
27fbb31144e8214	2017-09-18 10:50:56	PFCD-3537	PFCD-3537 cg on customerId should overwrite customer input value	
7335e6a328e0340	2017-09-06 08:04:35	PFCD-3537	PFCD-3537 simplify	
cd4ead0be8949a8	2017-09-06 07:48:01	PFCD-3537	PFCD-3537 add customer group input to quote root item + add config ui	
edee789d50fb593	2017-09-05 07:09:08	PFCD-3537	PFCD-3537 null customer must not overwrite customerId value from cg	
fa8f7f647265fd2	2017-09-05 06:37:11	PFCD-3537	PFCD-3537 set customerId on PR in case CG is set to a single customer	
ab8332ce9874d6c	2017-09-04 14:23:01	PFCD-3537	PFCD-3537 copy cg from qli to pricerecord	
436689142b581f3	2017-09-04 13:39:18	PFCD-3547	fix(Roles): 403 error when accessing Rebate Calculation Logic with limited permissions (PFCD-3547) To avoid the error, added permission to DATAMART_ACCESS to RM_FORMULAS_PERM. Serge will check for Cosmopolitan, if the code causing this is needed.	
463e42cfd750c08	2017-09-05 17:06:35	PFCD-3549	PFCD-3549 allow statuses in sandbox	
a2ceb69c7bbc822	2017-10-02 17:37:17	PFCD-3550	PFCD-3622 PFCD-3550 save current view state before switch to table view	
c35305ed735a19e	2017-10-01 05:16:08	PFCD-3550	PFCD-3622 PFCD-3550 prefs and state in q/ct/rba editors	
9a386a2b587c05d	2017-09-29 12:27:04	PFCD-3550	PFCD-3550 remove invalid commits	
bfd9d15508fafd0	2017-09-20 08:32:22	PFCD-3550	PFCD-3550 Contract default saved preferences not showing the correct view	
49900ddcd303ff3	2017-09-08 12:52:15	PFCD-3550	fix(ContractConsole): Contract default saved preferences not showing the correct view (#PFCD-3550)	
8c9c086edd40ad0	2017-09-11 13:47:59	PFCD-3551	PFCD-3551 simplify by using inner lexical scope	
c387d06ef68c06e	2017-09-11 13:01:07	PFCD-3551	PFCD-3551 unnecessary always true filter	
d2b0b7d68dbe38c	2017-09-11 12:33:02	PFCD-3551	PFCD-3551 filter by user's loginName in url	
6c998c111b3608b	2017-09-11 11:38:32	PFCD-3551	PFCD-3551 http endpoint for WF/fetch filtered by linked approvable	

67b30da904ef014	2017-09-05 11:02:30	PFCD-3553	PFCD-3553 New event type for PGI denial (e.g. ITEM_DENIED_PGI)	
ecfec9fab01fb71	2017-09-04 07:37:25	PFCD-3554	PFCD-3554 Fix some more XSS (stored & reflected) holes	
51355ac4e62e0bc	2017-09-11 11:38:30	PFCD-3562	PFCD-3562 Obtain context of LPG from matrix logic	
ff1fadc4af3f912	2017-09-06 08:28:07	PFCD-3565	PFCD-3565 Also add name field into EC loaddata requests for PX/CX (3a5e353)	
81bcccc950da600	2017-09-06 08:46:16	PFCD-3567	PFCD-3567 AddOrUpdateOutput for line item leads to Null Pointer Exception	
8deec3025c01f22	2017-09-06 09:38:35	PFCD-3569	PFCD-3569: Decrease #POS metrics logged	
510d846b43b0261	2017-09-06 09:54:20	PFCD-3570	PFCD-3570 import of entites with values in boolean fields fails	
b5dc949f70cc891	2017-09-07 15:19:25	PFCD-3583	PFCD-3583 allow all the other hc types to be used	
6de224e8c22256a	2017-09-07 12:58:15	PFCD-3583	PFCD-3583 regression in type field for highcharts	
365c40dbe739c1b	2017-09-12 06:42:36	PFCD-3587	PFCD-3587 Dashboard export to excel - scrambled order of data and header	
e44e25f5e9a64e5	2017-09-22 07:18:36	PFCD-3596	PFCD-3596 fix test on oracle	
505dfc1eeb9ca2e	2017-09-22 07:18:26	PFCD-3596	PFCD-3596 fix also PGI/PLI/MPLI/SIMI	
53908b78ca1f8c3	2017-09-22 07:18:14	PFCD-3596	PFCD-3596 api.find numerical sort works also on key fields	
6ced9b435917dec	2017-09-22 07:18:02	PFCD-3596	PFCD-3596 fix warn logging on multiple ams	
78147cb12affb41	2017-09-22 07:17:46	PFCD-3596	PFCD-3596 api.find fix sort on attributes on MLTV	
601177a6104106e	2017-09-28 06:50:17	PFCD-3607	PFCD-3607: allow grouping in small resultsets	
78193d7d553f88d	2017-09-18 13:43:28	PFCD-3607	PFCD-3607: PA-sim regression issues	
c382e1f7de65007	2017-09-21 12:38:47	PFCD-3609	PFCD-3609 Contract entitlement error message shown in wrong locale	
93b1a8dd7c731d2	2017-09-19 09:45:26	PFCD-3609	PFCD-3609 Adding some german labels	
edecaf75bb98272	2017-10-10 13:31:38	PFCD-3611	PFCD-3611: cherry-pick contract admin role	
7b5df609574f4fb	2017-10-12 14:24:30	PFCD-3614	PFCD-3614 - Email admin: Impossible to filter on subject field Fixed field type in data source for Subject	
f24e11d6db441f4	2017-10-04 08:29:19	PFCD-3614	PFCD-3614 - Email admin: Impossible to filter by email adress Fixed problem when also filtering with dates. NOTE: happens only if filtering is on server side, not in GWT client. Added test case.	
514aa2274dc42b5	2017-09-21 07:43:38	PFCD-3614	PFCD-3614 - Email admin: Impossible to filter by email adress	
59e6c607d6af0cb	2017-09-19 13:52:58	PFCD-3617	fix(PFCD-3617): PA: mass edit of field in DS fails with ACTION NOT ALLOWED	
dcc3ea581c4d483	2017-09-20 09:33:18	PFCD-3618	fix(PFCD-3618): DS: Warning message when creating new Data Source	
d8c51b3a9150bc4	2017-09-21 06:02:05	PFCD-3621	PFCD-3621 Setting correct mime type in case of raw XLS download (and macro enabled-template configured)	
a2ceb69c7bbc822	2017-10-02 17:37:17	PFCD-3622	PFCD-3622 PFCD-3550 save current view state before switch to table view	
c35305ed735a19e	2017-10-01 05:16:08	PFCD-3622	PFCD-3622 PFCD-3550 prefs and state in q/ct/rba editors	

13da87ce978dc4b	2017-09-30 19:59:27	PFCD-3622	PFCD-3622 apply prefs only for new q/ct/rba	
953c1d270b0fb18	2017-09-29 13:34:58	PFCD-3622	PFCD-3622 prefer preferences when quote is not saved	
6be4904711cfa22	2017-09-23 19:13:25	PFCD-3622	PFCD-3622 apply default prefs after switch back from table view	
84c72177a573af9	2017-09-22 10:29:47	PFCD-3622	PFCD-3622 Overview in Quote/RBA - apply preferences automatically only after save quote/rba	
c49d11ca5369540	2017-09-20 11:07:38	PFCD-3623	PFCD-3623 - Rebate Calculations - User needs to be General admin else error Added missing permission Permission.JOBSTATUSTRACKER_FETCH to REBATE MANAGER ROLE.	
ed8fb5b1c610973	2017-09-21 07:42:39	PFCD-3626	PFCD-3626 Quote - Result groups - shows duplicate values	
6b4925e5e4fb6f4	2017-10-04 08:17:57	PFCD-3648	PFCD-3648 ResultMatrix auto-fit not enabled anymore by default - now also for dashboards again (take 2)	
115b7dfff1f2057	2017-10-04 06:09:51	PFCD-3648	PFCD-3648 ResultMatrix auto-fit not enabled anymore by default - now also for dashboards again	
5364e6eb1437dda	2017-09-27 06:09:20	PFCD-3648	PFCD-3648 ResultMatrix auto-fit not enabled anymore by default	
09fab722bbaed85	2017-10-04 06:45:22	PFCD-3650	PFCD-3650 Events for Quote object - also do cleanup again	
ea2f189780fa5ab	2017-10-02 09:56:11	PFCD-3650	PFCD-3650 Events for Quote object	
6dc1569d5141b62	2017-10-02 07:23:12	PFCD-3652	PFCD-3652 Fix a NPE	
c1ff9233c13f7fb	2017-09-28 10:04:52	PFCD-3655	PFCD-3655 ResultMatrix shows distorted columns in special cases	
89553c1e3ae238b	2017-10-12 06:27:33	PFCD-3656	PFCD-3656 Minor fix	
c1286046711f9dd	2017-10-05 09:47:12	PFCD-3656	PFCD-3656 Allow more granular finetuning of SAML requests - UI parts	
d905a7a985ca5b0	2017-09-28 11:04:59	PFCD-3656	PFCD-3656 Allow more granular finetuning of SAML requests	
4bb06475bcc14c3	2017-10-02 17:42:45	PFCD-3660	PFCD-3660 fix SmartGwtUtil.getFilterBuilderCompatibleCriteria use SmartGWT's JSOHelper.createJavaScriptArray() instead of GWT's JsArrayNumber.createArray() to avoid errors due to missing getUniqueItems() function	
c1438a18987efd4	2017-10-11 12:15:20	PFCD-3666	PFCD-3666 Make sure also items added by header PRE logic are marked dirty all the way up	
31d548674a02418	2017-10-04 06:02:47	PFCD-3666	PFCD-3666 Items added by header logic in folders (i.e. not root level) are not calculated immediately	
ecfab5139413f01	2017-10-04 13:59:15	PFCD-3671	PFCD-3671 XSS fixes	
84cb96593538ac7	2017-10-04 12:29:08	PFCD-3671	PFCD-3671 XSS fixes	
1c5dd580d47ad86	2017-10-12 15:12:56	PFCD-3675	PFCD-3675 table view xss	
7204f32d0834d83	2017-10-09 08:53:06	PFCD-3675	PFCD-3675 XSS holes - 23nd shot	
1129f1677551745	2017-10-07 15:51:21	PFCD-3675	PFCD-3675 XSS holes - 22nd shot	
6f5d135d6e5e477	2017-10-06 15:34:38	PFCD-3675	PFCD-3675 XSS holes - 21st shot	
fb9f5605420885d	2017-10-06 15:22:56	PFCD-3675	PFCD-3675 XSS holes - 20th shot	
1a7db5c857e9efc	2017-10-06 14:40:09	PFCD-3675	PFCD-3675 Cosmetic fixes	
47b2db8b3585cb6	2017-10-06 14:05:15	PFCD-3675	PFCD-3675 XSS holes - 19th shot	

364e41db1e5a3ca	2017-10-06 13:58:15	PFCD-3675	PFCD-3675 XSS holes - 18th shot	
a3b48d7579901e8	2017-10-06 12:53:05	PFCD-3675	PFCD-3675 XSS holes - 17th shot	
e13e41bd40950f8	2017-10-06 11:37:35	PFCD-3675	PFCD-3675 XSS holes - 16th shot	
87414a5d2cf3f1c	2017-10-06 10:21:24	PFCD-3675	PFCD-3675 Re-enable ref links	
7763203ca6bde9d	2017-10-06 09:42:15	PFCD-3675	PFCD-3675 XSS holes - 15th shot	
2aaf4ab67d8a75b	2017-10-06 09:06:21	PFCD-3675	PFCD-3675 XSS holes - 14th shot	
d34a57d0dd323d2	2017-10-06 08:46:17	PFCD-3675	PFCD-3675 XSS holes - 13th shot	
2dfe4e4ad5efe54	2017-10-06 08:27:41	PFCD-3675	PFCD-3675 XSS holes - 12th shot	
c22e6559cd72c42	2017-10-06 07:41:48	PFCD-3675	PFCD-3675 XSS holes - 11th shot	
24fd3515b106b86	2017-10-06 07:00:52	PFCD-3675	PFCD-3675 XSS holes - 10th shot	
945f653cff4e16d	2017-10-06 06:51:28	PFCD-3675	PFCD-3675 XSS holes - 9th shot	
349e618aa54a283	2017-10-06 05:26:22	PFCD-3675	PFCD-3675 XSS holes - 8th shot	
179ec05c1c08510	2017-10-05 21:39:16	PFCD-3675	PFCD-3675 XSS holes - 7th shot	
3c1d03e4fd49a13	2017-10-05 19:00:35	PFCD-3675	PFCD-3675 XSS holes - 6th shot	
13fe455b9bf08c7	2017-10-05 18:31:48	PFCD-3675	PFCD-3675 XSS holes - 5th shot	
31f7ee63990de95	2017-10-05 18:22:59	PFCD-3675	PFCD-3675 XSS holes - 4th shot	
4c82ba225e0ae56	2017-10-05 16:47:48	PFCD-3675	PFCD-3675 XSS holes - 3rd shot	
f6f9b45fed5621b	2017-10-05 15:00:41	PFCD-3675	PFCD-3675 XSS holes - 2nd shot	
7227e93f39a6566	2017-10-05 14:38:33	PFCD-3675	PFCD-3675 XSS holes - first shot	
db13501f39e1791	2017-10-09 06:57:07	PFCD-3676	PFCD-3676 Items lost during LPG calculation	
f9b2346c40e3761	2017-10-09 10:32:34	PFCD-3678	PFCD-3678 - On Contracts the validation 'startDate < endDate' happens on addLineItem but NOT on recalculate Added validation on "Recalculate" of Quote and Contract. Also improved validator to validate immediately after "End Date" on Contract is changed, resp. Expiry Date on Quote.	
95ca93906404955	2017-10-06 13:09:04	PFCD-3680	PFCD-3680 Labels in Visibility settings (Save chart)	
600e4921f258d8e	2017-10-09 11:19:21	PFCD-3690	fix(PFCD-3690): DatamartFilterBuilderUserEntry: in AND-filter, only the first clause is preserved	
8c3df5ce0462666	2017-10-10 14:50:40	PFCD-3691	PFCD-3691 attachments in contract console	
c91efea95cbc91c	2017-10-10 13:44:47	PFCD-3691	PFCD-3691 only Contracts Admin can delete/reupload attachments to readonly CT	
4756985c837dbfe	2017-10-10 11:07:31	PFCD-3692	PFCD-3692 ability do disable attaching of attachments on submit	
a862ee02b041b31	2017-10-20 13:09:25	PFCD-3694	PFCD-3694 DE language review	
ad5a5b024a8b19f	2017-10-20 13:07:31	PFCD-3694	PFCD-3694 language review	

deaec2905800034	2017-10-10 06:16:41	PFCD-3694	PFCD-3694 Option to make TFA mandatory on partition level	
23f9231111aa85c	2017-10-29 19:14:47	PFCD-3714	PFCD-3714: performance tweak: do not openA;ll FilterTree	
e05dfdd46ebce88	2017-10-12 10:22:35	PFCD-3717	PFCD-3717 Preload distributed result items to improve throughput	
68b19667beeb8d8	2017-10-16 17:08:25	PFCD-3726	PFCD-3726: api.datamartFilterBuilderUserEntry - not able to enter filter label with spaces	
afafd1ebe5dcc7c	2017-10-23 10:59:22	PFCD-3727	fix(t PFCD-3727): PA filter builder not working in dashboards	
9b3adb82ec276fd	2017-10-16 17:19:34	PFCD-3727	fix(PFCD-3727)	
56de80dee00915c	2017-11-02 12:49:22	PFCD-3731	fix(PFCD-3731)	
8f648929e5526ad	2017-10-18 18:50:11	PFCD-3732	PFCD-3732 deny should also respect order of groups	
89ec0f6eee839a0	2017-10-18 05:04:53	PFCD-3732	PFCD-3732 order of group fromw wf definition rather than from user definition	
80b9f976e29768c	2017-10-17 20:37:03	PFCD-3732	PFCD-3732 avoid the same groups in history	
25c1d91ec003273	2017-10-17 19:41:10	PFCD-3732	PFCD-3732 allow single user to approve multiple times, in case of userGroup approval and multiple groups	
db62bbc2786ec98	2017-10-18 08:51:34	PFCD-3734	PFCD-3734 publish wf step history in wf info	
97b00e985b3c77a	2017-10-18 13:08:30	PFCD-3737	PFCD-3737 approval window - comment field - stalled cache	
4c9532665570442	2017-10-18 13:11:10	PFCD-3738	PFCD-3738 store step comments in wf history	
0d9b19fb0d40298	2017-10-19 11:01:27	PFCD-3741	PFCD-3741 css bump	
b6e9d8adb70d6c6	2017-10-19 08:43:49	PFCD-3741	PFCD-3741 - optimise prerender - fixed autoadjust loop	
118bb607a5760b7	2017-10-19 15:35:55	PFCD-3745	PFCD-3745 - User can not log in when he is member of 1000+ groups (ORA-01795) Removed wrong import!	
16438db898c4581	2017-10-19 15:27:45	PFCD-3745	PFCD-3745 - User can not log in when he is member of 1000+ groups (ORA-01795) Hot fix + improved IT to cover both cases (number of groups below 1k and over 1k)	
86c6711f112e197	2017-10-19 15:27:38	PFCD-3745	PFCD-3745 - User can not log in when he is member of 1000+ groups (ORA-01795) Created IT for to reproduce this on Oracle.	
9a37d692987452b	2017-10-20 09:25:46	PFCD-3750	PFCD-3750 Prevent fieldType toggling back to "unknown" in LPG	
631050a12068cab	2017-11-02 13:31:55	PFCD-3770	PFCD-3770 - UI issue in latest Chrome code clean up (imports etc)	
cf0300485892aa6	2017-10-30 17:07:13	PFCD-3783	fix(PFCD-3783): query on rollup ; col not found (spaces)	
d76c05d1bfabc6e	2017-11-01 09:01:05	PFCD-3784	PFCD-3784 hide also all dynamically generated tabs	
f3a94047b782316	2017-11-02 11:37:39	PFCD-3800	PFCD-3800 - Users marked with â€œis deletedâ€ are showing up in download report for user audit. Change the report to exclude "deleted" users.	
d8715e13777a990	2017-11-16 07:55:57	PFCD-3837	PFCD-3837 Fix NPE from tests	
8430f99ee73f353	2017-11-16 07:55:31	PFCD-3837	PFCD-3837 AttributeContext not cleared for slave threads	
a4382af83b90be8	2017-11-28 14:06:57	PFCD-3868	PFCD-3868 ORA-01795 more than 1000 values in IN clause	
51f8348524d9c36	2017-10-12 11:00:10	PFSFDCQC-434	PFSFDCQC-434 - sorted	

4439b30ad393e72	2017-10-12 10:30:39	PFSFDCQC-434	PFSFDCQC-434 - merging into core localisations	
3dbb88d802d2519	2017-10-23 13:24:15	PFSFDCQC-460	PFSFDCQC-460 - extra translation	
8a5a2b572883513	2017-09-18 10:11:57	PR-116	PFCD-3104: minor mods following PR-116 feedback	

Mojito 3.2 Changelog

Commit	Date	Issue	Message
98dc4a5590223fb	2017-05-30 08:56:36	PFCD-2339	[EC] Fix for PFCD-2339 & PFCD-3148
33326d47bc88a5f	2017-05-05 12:25:26	PFCD-2837	fix(tree editor) #PFCD-2837
19b062023cfc423	2017-06-20 11:14:20	PFCD-2899	PFCD-2899 workflow history button in PGI grid
dc74690ec3ecd63	2017-05-02 07:55:37	PFCD-3072	PFCD-3072 fix ResultMatrix deserialization
05c5fed363cf566	2017-05-15 16:04:57	PFCD-3112	fix(SavedChart) chart owner cannot edit/delete PUBLIC chart (#PFCD-3165) regression due to PFCD-3112
8247c9cb050c1dd	2017-05-09 12:33:27	PFCD-3112	fix(SavedChart UI) {en/dis}able of save and delete buttons (#PFCD-3112) In load/save dialog of ChartEditor, make sure the save and delete buttons are show and enabled at the right time (i.e. if chart is public, or user is either owner of the chart or admin)
c65ffc713c0996e	2017-05-05 14:23:25	PFCD-3112	fix(SavedChart) Saved Charts Admin now also able to edit (#PFCD-3112) To be consistent with the first commit for this issue, a chart admin should not only be able to delete, but also to edit other users' charts. Since the admin would then be the lastUpdateBy user, the owner of the chart is now checked using createdBy field instead of lastUpdateBy.
2bb009647484eea	2017-05-05 12:13:38	PFCD-3112	fix(typos and warnings) relating to PFCD-3112
f1a7b39f9bb4d5c	2017-04-28 15:19:45	PFCD-3112	fix(SavedChart) Saved Charts Admin unable to delete chart (#PFCD-3112) Up to now only the chart's owner was able to delete private or published charts. Now "Saved Charts Admin" role can also delete, even if he's not owner.
e400bad15e0ce4f	2017-05-02 14:27:18	PFCD-3113	PFCD-3113 setting IDs breaks the app, use styles please
68bbbdb4be7e5c7	2017-05-02 06:27:31	PFCD-3114	PFCD-3114 useServerPrepStmts=true in mariadb driver causes leads to failed prep stmts if db server caches not adjusted
64359ffa6883f7b	2017-05-04 10:49:45	PFCD-3115	fix(PA query cache files not deleted) #PFCD-3115
98939442a05e2b3	2017-05-03 06:11:25	PFCD-3118	PFCD-3118 Manual override do not works for "Pricelist (Editor)" role
9bb366109cfcfae	2017-05-02 14:19:02	PFCD-3119	#PFCD-3119 Formula Elements with element group conditions may not be compiled during save
28028e7f952b084	2017-05-03 12:54:40	PFCD-3120	(backport) #PFCD-3120 Enable recalculation of specific manual pricelists (i.e. also ones valid in the future or older revisions) Fix GWT compatibility issue
0adb1a0c3e29bf4	2017-05-03 09:49:48	PFCD-3120	(backport) #PFCD-3120 Enable recalculation of specific manual pricelists (i.e. also ones valid in the future or older revisions)
e6eda585ec9f5d5	2017-05-03 09:37:50	PFCD-3120	(backport) #PFCD-3120 Enable recalculation of specific manual pricelists (i.e. also ones valid in the future or older revisions)
3394d7a099abad8	2017-05-04 12:06:01	PFCD-3124	PFCD-3124 itemTable actions visibility
04909e57ca77680	2017-05-04 09:02:45	PFCD-3124	PFCD-3124 mass delete and mass edit should not be editable when agreement is readonly
5d1af6afc694404	2017-05-04 09:42:41	PFCD-3126	fix(PA Data Manager role) needs JST fetch permission (#PFCD-3126)
279aacd1ed81d8a	2017-05-09 12:38:09	PFCD-3128	PFCD-3128 delete button enabled in non-editable states in agreement table view
b5e1808036246a0	2017-05-11 13:01:52	PFCD-3129	fix(FlexChartTemplatesConfigurationPage) (#PFCD-3129) Make sure not to save an empty default base template

97bc5002693b9d1	2017-05-11 13:01:51	PFCD-3129	fix(FlexChartsUtil) avoid JSO syntax error (#PFCD-3129) Made loadFlexChartTemplates() resistant to empty templates, as this was freezing the whole app just after login.
1655dcebbf46782	2017-06-09 16:34:23	PFCD-3131	PFCD-3131 fix date/time display in DataLoads list grid Was displaying UTC time, which was confusing since right hand side configuration pane shows local time zone.
cbbe0c4768ca648	2017-06-09 14:27:33	PFCD-3131	PFCD-3131 fix for source and target tabs of the DL configuration pane Was due to the way we update the dataload record to refresh the grid state. Now only updating the relevant attributes, i.e. grid fields.
2a5066f086c1c5c	2017-06-09 09:47:16	PFCD-3131	PFCD-3131 fixed an unrelated NPE while working on the main issue When canceling a DL task, the console was showing a stack trace. The functionality was working anyway, which is why I didn't bother creating a separate JIRA.
b831e2d02fd3363	2017-06-09 09:42:28	PFCD-3131	PFCD-3131 DataLoad configuration pane empty after running a DL Was due to a BooleanItem where getValueAsBoolean() would sometimes return null even though it had been set.
36c64141cf59a0d	2017-05-08 06:37:50	PFCD-3142	#PFCD-3142 OX Move with custom mapping does not move Meta data accordingly
d0dbb9aa5918505	2017-05-09 06:50:53	PFCD-3143	#PFCD-3143 Pricelist batch item delete does not work on Matrix-PL
440bdb4706b4d58	2017-05-11 09:39:44	PFCD-3144	fix(Formula Test Window) : remember selected inputs and parameters between test drives (#PFCD-3150 #PFCD-3144)
b58bd970ae2219f	2017-05-15 05:50:11	PFCD-3147	PFCD-3147 Adding JST.fetch permission to a number of roles
98dc4a5590223fb	2017-05-30 08:56:36	PFCD-3148	[EC] Fix for PFCD-2339 & PFCD-3148
440bdb4706b4d58	2017-05-11 09:39:44	PFCD-3150	fix(Formula Test Window) : remember selected inputs and parameters between test drives (#PFCD-3150 #PFCD-3144)
26f6f3eedf9e428	2017-05-11 15:56:25	PFCD-3151	PFCD-3151 empty tableView when switching between agreements with different structure
544c9506861f256	2017-05-30 13:13:14	PFCD-3153	PFCD-3153 TableView: add missing field name escaping
594fa16a40fd3d0	2017-05-30 11:18:00	PFCD-3153	PFCD-3153 fix unittest
dbdda55d5fcd541	2017-05-30 11:04:09	PFCD-3153	PFCD-3153 TableView is not shown when there are inputs with / in name
1d81afce03b8fb1	2017-05-16 13:47:52	PFCD-3156	PFCD-3156 improve logging
c4c0b5c084da523	2017-05-16 13:20:08	PFCD-3156	PFCD-3156 not fixed but bounded partition threadpool
f1a7b30765f800c	2017-05-15 19:07:56	PFCD-3156	PFCD-3156 remove backgroundtaskprocessor from dagger component
ca06a6575dde2c5	2017-05-15 19:02:30	PFCD-3156	PFCD-3156 final and pool size
c127c267d8e6238	2017-05-15 14:37:56	PFCD-3156	PFCD-3156 try to continue after per-partition batch is processed
75a5c9fedaf64d5	2017-05-15 13:52:57	PFCD-3156	PFCD-3156 remove obsolete settings
77be22faa166f6d	2017-05-15 13:41:43	PFCD-3156	PFCD-3156 avoid arraylist realloc
2e5aaa6291c79cc	2017-05-15 13:08:44	PFCD-3156	PFCD-3156 remove grouping by target url
46573dac4722547	2017-05-15 12:39:05	PFCD-3156	PFCD-3156 per-partition thread should process only task that belong to the partition
e1d0c3cb22b5d22	2017-05-15 11:50:51	PFCD-3156	PFCD-3156 thread only for partition with eventUrl set
27965f73f6c6565	2017-05-15 11:18:02	PFCD-3156	PFCD-3156 inttest threadpool
4cccae1ad72d1d9	2017-05-15 11:08:25	PFCD-3156	PFCD-3156 inttest adjustments
ca26f2ed06e6482	2017-05-15 10:39:25	PFCD-3156	PFCD-3156 thread names and log levels

8165b23a9b18f5a	2017-05-15 10:24:04	PFCD-3156	PFCD-3156 whitespace in config
55ed34dfe399da5	2017-05-15 10:20:21	PFCD-3156	PFCD-3156 per partition event processing
3af9bce4d65abbe	2017-05-15 10:23:21	PFCD-3158	PFCD-3158 Also fix correct archive and export for PX in backend
c1e1373ef05d8a9	2017-05-12 12:51:41	PFCD-3158	#PFCD-3158 Data Archiving does not work on Extensions
f0a93379bf0d3ce	2017-05-17 10:50:15	PFCD-3164	PFCD-3164 empty validation error in Q/CT/RA
cdbb8a23cd48f73	2017-05-17 07:45:09	PFCD-3164	#PFCD-3164 Empty error messages appearing on various partitions
a860b4d00684fbf	2017-05-16 08:18:12	PFCD-3164	PFCD-3164 Extend default error handler to help troubleshooting
05c5fed363cf566	2017-05-15 16:04:57	PFCD-3165	fix(SavedChart) chart owner cannot edit/delete PUBLIC chart (#PFCD-3165) regression due to PFCD-3112
d65bac3a169ed23	2017-05-15 14:44:36	PFCD-3169	fix(ResultMatrixPopup): Some dashboards (ResultMatrix) don't show any data (#PFCD-3169) We found that the problem is linked to the dashboard portlets, that contain this code in Formula Groovy element: matrix.setEnableClientFilter(true) If the "client filter" is enabled in the matrix grid, it does not show any data. Problematic line is detailsGrid.setFilterLocalData(true); It should only be used when data is provided via setData() to grid. We don't use this approach anymore.
5fe22dd53a63e41	2017-05-19 15:25:33	PFCD-3175	fix(ChartEditor) comparison charts failing in IE11 (#PFCD-3175) For some reason SGWT implementation of System.identityHashCode(this) way too easily returned the same value for two distinct instances of the same class in IE11.
5fb8e7d1c5ca679	2017-05-19 06:06:49	PFCD-3182	PFCD-3182 Also apply redis timeout to use existing retry mechanism for HZ timeouts
c665b676cad3239	2017-05-18 09:49:29	PFCD-3183	PFCD-3183 TableView: mass add, unable to add multiple lines of same contractType
0b0f637582b4d26	2017-05-19 12:27:37	PFCD-3184	PFCD-3184 TableView: fix numerical type handling
aefe96de30c4dc7	2017-05-18 12:32:12	PFCD-3184	PFCD-3184 TableView: mass adding input with REAL datatype in parameter details leads to expected Integer instead of String error
9294345a0269d9b	2017-05-19 12:47:57	PFCD-3190	PFCD-3190 Product Quick search - result display problem if other elements hidden
e0cd451b0f100a9	2017-05-19 12:29:10	PFCD-3196	PFCD-3196 TableView: mass add column mapping switching can be slow
fc24c14bd1261fa	2017-05-19 13:24:19	PFCD-3198	#PFCD-3198 Create "ping" command
bdfdb3fdd7d08d3	2017-05-23 05:52:40	PFCD-3199	PFCD-3199 Optionally skip auto-approval if item is dirty
581aeadd25d5f2e	2017-05-23 08:44:19	PFCD-3206	PFCD-3206 fixing broken edits in mass add grid by previous commit
0e7ec6c3b05066c	2017-05-22 20:11:57	PFCD-3206	PFCD-3206 editing values in mass add view has no effect
0b99ddd07e02606	2017-05-22 14:23:08	PFCD-3206	PFCD-3206 TableView: mass add and mass edit type handling (avoid coercion to string in all cases)
125a3d7b29ebbd7	2017-05-23 14:34:47	PFCD-3210	PFCD-3210 TableView: refresh after mass add commit fix
8abf02930360a20	2017-05-23 12:40:56	PFCD-3210	PFCD-3210 TableView: outputs not refreshed after price
f42aec6651f77c1	2017-07-10 10:42:19	PFCD-3222	PFCD-3222 YAxis aggregation was ignored by ChartBuilder for ScatterChart
ad3e76f8f01caca	2017-06-29 17:43:53	PFCD-3222	PFCD-3222 Count Distinct was not supported at all in ChartBuilder Hint Now making sure to add .withAggregation('COUNTDISTINCT') where needed, i.e. in nearly all chart types on various axes and measures.
5a0d6c01bb548e8	2017-06-29 14:01:00	PFCD-3222	PFCD-3222 Fix a few warnings

3bf7b33663b3e05	2017-05-30 08:30:13	PFCD-3226	PFCD-3226 - submit is hidden for critical alerts
295f76a224fb57f	2017-06-13 10:14:27	PFCD-3228	fix(DataProfileResult not whitelisted) #PFCD-3228
09bc60668b48b1f	2017-05-26 14:16:21	PFCD-3228	PFCD-3228: added list formatting to Matrix2D public + DatamartContext javadoc
33d907977f2b92f	2017-05-26 13:36:45	PFCD-3228	PFCD-3228: Matrix2D public javadoc + corrections in DatamartContext & Query. Also pulling in new DataProfilerResult as part of the documentation effort.
577bc8603197c78	2017-05-29 08:55:04	PFCD-3230	PFCD-3230 Fix XLS export of long/large numbers (anything that's bigger than int, i.e. is long)
cffef97b0fcd658	2017-05-30 13:14:35	PFCD-3232	PFCD-3232 Tableview: Cannot edit Customers/Products on LI level
f82185cb27cee74	2017-08-08 12:51:50	PFCD-3233	PFCD-3463 Column name containing <space> breaks ResultMatrix click event Relates to PFCD-3233 which introduced a possible discrepancy between a field's name and a field's title (logical versus display name), hence a difference between the matrix's column name and a grid record's attributes.
e41efcfe9f5d7e2	2017-06-07 11:04:35	PFCD-3233	fix(ResultMatrixPopup): Dashboard doesn't display some data + Excel export broken (PFCD-3233) Improved performance of JavaScriptUtil.makeValidJSIdentifier(String name)
6d3522828696877	2017-06-07 10:12:11	PFCD-3233	fix(ResultMatrixPopup): Dashboard doesn't display some data + Excel export broken (PFCD-3233) Fixing problem when some data from JSON was not displayed in dashboard populated from Matrix Result. Reason that the name of the FieldGridField was not a valid JS identifier. So did a minimal fix to workaround this problem in JavaScriptUtil.makeValidJSIdentifier(...) method to fix this problem. I will file a separate improvement to complete makeValidJSIdentifier(...) method because the current implementation is not really complete.
51c09db77633819	2017-06-01 09:11:42	PFCD-3244	fix(cannot cancel PA DL due to ActionStatus dropped from cache): #PFCD-3244
88dc5109c1d1a3e	2017-06-09 12:16:32	PFCD-3279	PFCD-3279 PX and CX filter in import/export/archive
275353c07899aa8	2017-06-13 07:25:23	PFCD-3290	PFCD-3290 workflow approved email not sent
6528500fc5369f4	2017-06-21 12:29:44	PFCD-3297	PFCD-3297 -> test refactoring
9d0586fc6246a37	2017-06-19 12:43:08	PFCD-3297	PFCD-3297 -> refactoring from code review
5bf207e45c7fcf0	2017-06-14 13:34:08	PFCD-3297	PFCD-3297 -> implemented api endpoint
09f9dd336e3aaf4	2017-06-14 12:28:02	PFCD-3297	PFCD-3297 -> implemented api for streaming of DCR changes
e987878016d7df0	2017-07-12 08:50:43	PFCD-3303	PFCD-3303 Stepped lines still failing if X axis was showing only years "2017" was read as a number (i.e. index of an x axis value) instead of as a string (i.e. label on the x axis).
060b350802f7b54	2017-07-06 16:56:06	PFCD-3303	PFCD-3303 sample X value when adding stepped line points in time charts
ce604e775aa8dc6	2017-07-03 17:27:37	PFCD-3303	PFCD-3303 fix a small low level highcharts exception when zooming
bf239c2972f83fc	2017-06-23 16:21:58	PFCD-3303	PFCD-3303 Reverted the part that's been cherry picked to 3321
36e54c6f6f094a3	2017-06-21 16:10:57	PFCD-3303	PFCD-3303 Stepped Line support for TimeSeriesScatter and Datetime charts
fe4326d8684e261	2017-06-21 10:08:28	PFCD-3303	PFCD-3303 Stepped Line support for TimeSeriesChart
55d44084a0c8b39	2017-06-29 10:59:22	PFCD-3306	PFCD-3306 PA FetchData now filters out undesired fields if requested Now checking if a fields list was specified in request.
784eb61387a937e	2017-07-03 09:48:19	PFCD-3307	PFCD-3307 Fix validation of PRs to allow concurrent setting of sku and PG field if they are actually pointing to same thing
132fd104dde5e4f	2017-06-23 08:00:00	PFCD-3307	PFCD-3307 Copy SKU / CustomerId to dedictaed PR fields if PG/CG essentially defines only that
05c9bd05368218e	2017-07-07 14:52:07	PFCD-3311	PFCD-3311 Make sure PASchemaManager is initialized in ChartPortlet For a user with only PA_QUERY role, it's not done at login time.

f21eb64f0c39f90	2017-06-21 09:22:12	PFCD-3311	PFCD-3311 Hide load/save buttons if user can't fetch the chart tree Avoids warning in dashboard charts if user has only PA Run Query role.
fa2fbe32a9d790b	2017-06-20 08:08:27	PFCD-3312	bug(Main Menu): Hide the Price Analyzer menu if user has only the "PA Run Query" role (PFCD-3312) User that does not have any other PA role than Role.PA_QUERY will not see PA menu at all.
7669de838dc3b16	2017-06-26 17:03:58	PFCD-3321	PFCD-3321 fix zooming in TimeSeriesScatter and Bubble charts Generalized the fix that was previously applied only to ScatterChart. Also includes an additional fix for a JS undefined exception in getHighchartDefinition() and a missing icon for TimeSeriesScatterChart.
f689526e52ba6c5	2017-06-23 09:12:38	PFCD-3321	PFCD-3321 fix zooming in ScatterChart Use ViewPort info for min/max values when building the Highchart definition.
4c67aae231c42d7	2017-06-22 08:46:08	PFCD-3321	PFCD-3321 stop ignoring first and last x coordinate in stepped lines
4f902587a890226	2017-06-23 10:53:13	PFCD-3325	PFCD-3325 Matrix price grid item is not distinguished by key2 (significatField array is wrong)
d52609400392bb9	2017-06-28 05:20:56	PFCD-3327	PFCD-3327 Manual Price List deletion - isDeleted=1 but still visible in grid
c575cf38012d6da	2017-07-10 10:42:18	PFCD-3331	PFCD-3331 Cumul. Contrib. and others not supported in ChartBuilder Hint Forgot one BubbleChart case in previous commit in this same branch.
99759ab6a3f411c	2017-06-30 10:00:34	PFCD-3331	PFCD-3331 Cumul. Contrib. an others not supported in ChartBuilder Hint Now simply adding .withAggregation(<aggregation name>) where needed, i.e. in nearly all chart types on various axes, measures and limitTo.
86dc425668bed13	2017-07-03 15:26:28	PFCD-3334	fix(ResultMatrixPopup): Dashboard doesn't display correct data received in JSON (PFCD-3334) DataFetchMode on grid must be set during the component creation, not after that.
6bd0729ec9adb33	2017-08-14 16:41:12	PFCD-3340	PFCD-3340 - table scroll performance boost - backported to Mochito SE (PFCD-3397)
76b3b85cf85c01e	2017-06-30 09:16:43	PFCD-3341	PFCD-3341 Option to upload templates also for matrix flavours of PL & PG
e33404860624e5e	2017-07-10 14:00:24	PFCD-3348	fix(web-CalculableLineItemCollection): Detail of Quote is not displayed in Classic UI if created via JC Custom UI (PFCD-3348) Made Classic UI more robust to handle opening quotes containing input parameter with null name.
29b4e1c524eb9b8	2017-07-11 08:02:39	PFCD-3367	PFCD-3367 vertical auxiliary lines were not shown in the chart legend
24c87bbf87bb8af	2017-07-20 15:38:33	PFCD-3370	PFCD-3393 Charts fix: vertical aux lines regression due to PFCD-3370 Vertical auxiliary lines would not show up anymore when loading a chart or building it in groovy logic.
2867286df6fcb2e	2017-07-12 12:07:09	PFCD-3370	PFCD-3370 sample X value when adding a vertical auxiliary line
a3dc42a755a9445	2017-07-12 12:06:50	PFCD-3370	PFCD-3370 Add vertical line support to time based series
6cf0ed76f872a48	2017-07-13 12:55:42	PFCD-3375	PFCD-3375: IT-fix
22870640bb18546	2017-07-12 19:59:53	PFCD-3375	PFCD-3375: PA: createIndex option in system$redeployfcs command
24c87bbf87bb8af	2017-07-20 15:38:33	PFCD-3393	PFCD-3393 Charts fix: vertical aux lines regression due to PFCD-3370 Vertical auxiliary lines would not show up anymore when loading a chart or building it in groovy logic.
6bd0729ec9adb33	2017-08-14 16:41:12	PFCD-3397	PFCD-3340 - table scroll performance boost - backported to Mochito SE (PFCD-3397)
17d576a4d116f5c	2017-07-20 14:08:14	PFCD-3400	PFCD-3400 ChartBuilder fixes for additional measures in various charts impacted charts: Histogram, Scatter and TimeSeriesScatter
1e52e369beaa01d	2017-07-31 08:10:06	PFCD-3412	PFCD-3412 Waterfall Comparison chart shows duplicated bars in dashboard Fixed by not refreshing dashboard charts on tab switch anymore, as we have an explicit button for that in the tab's label
1d53348ed7dd967	2017-08-10 09:29:25	PFCD-3429	PFCD-3429 check for index drop exception's sql state ... better than doing string comparison of the error message
1dc34dc9d1d2df0	2017-08-09 16:39:11	PFCD-3429	PFCD-3429 Catch exception if index drop fails because it doesn't exist Forgot one of the dropIndex() calls in the first commit
1a69b05ad4114ff	2017-08-09 14:05:35	PFCD-3429	PFCD-3429 Catch exception if index drop fails because it doesn't exist
1926ec2be498f4c	2017-08-03 12:04:16	PFCD-3441	PFCD-3441 Also fix alternate code paths

71b64d9fcc33902	2017-08-01 14:56:34	PFCD-3441	PFCD-3441 LPG partial calculation not scheduled when it is scheduled from logic (calculation flow) and PG in "wrong" status
ac0cfbb955ac7df	2017-08-01 14:43:44	PFCD-3441	PFCD-3441 LPG partial calculation not scheduled when it is scheduled from logic (calculation flow) and PG in "wrong" status
a11b9f6d7be0366	2017-08-07 15:07:54	PFCD-3445	PFCD-3445: code review
82eddda808c6588	2017-08-03 09:20:55	PFCD-3445	PFCD-3445 Slow Price list opening performance
19b256f703c0dd3	2017-08-08 12:21:38	PFCD-3452	PFCD-3452: repair infinite loop in iterative computation of live matrix price grid - distributed mode
867f75b9ed9264f	2017-08-08 11:04:34	PFCD-3452	PFCD-3452: repair infinite loop in iterative computation of live matrix price grid
f82185cb27cee74	2017-08-08 12:51:50	PFCD-3463	PFCD-3463 Column name containing <space> breaks ResultMatrix click event Relates to PFCD-3233 which introduced a possible discrepancy between a field's name and a field's title (logical versus display name), hence a difference between the matrix's column name and a grid record's attributes.
52b23984060e7eb	2017-08-14 13:30:10	PFCD-3475	PFCD-3475 - full refresh of grid instead of partial refesh
5e5112602395d87	2017-08-21 07:15:23	PFCD-3476	PFCD-3476: test review
43f06c33df43d31	2017-08-18 07:08:04	PFCD-3476	PFCD-3476: tests check that a new JST is scheduled when parameters differ
79715729e9e1ef8	2017-08-10 09:49:55	PFCD-3476	PFCD-3476: schedule a new JST also when parameters differ

Mai Tai 3.1 pricefx-server Changelog

- pricefx-server ChangeLog Issue Specific
- pricefx-server ChangeLog All changes

pricefx-server ChangeLog Issue Specific

Commit	Date	Issue	Message
64e1e1e53110f8b	2016-11-29 09:42:08	PFCD-2205	PFCD-2205 clouconvert agreement token can't be read correctly on non-remote schemas
c53d68ec9c57dd7	2016-12-01 14:17:18	PFCD-2282	PFCD-2282 Handle errors in the feeder logic of PA Calculation DLs Both formula errors and any other feeder exception were ignored. Also added JST progress info updates.
65e7a2747e63da4	2016-12-21 13:07:41	PFCD-2514	Revert "PFCD-2514 allow run with older server" This reverts commit 4af0634a7aea85b22a7fefe0ea40eca6f521c60b.
4af0634a7aea85b	2016-12-21 13:04:27	PFCD-2514	PFCD-2514 allow run with older server
f091c4de19dfc03	2016-12-14 12:44:04	PFCD-2514	PFCD-2514 pdf export button available only when clouconvert enabled and xls template present
4b0b0b13f0f245d	2016-12-13 15:15:12	PFCD-2519	PFCD-2519 PA calculations: formula errors not resulting in ERROR status
e04ad37d486feb2	2016-12-07 17:44:22	PFCD-2527	PFCD-2527 PA: TOSTRING expression was not returning String type
98251c8907951e7	2016-11-26 15:36:47	PFCD-2536	Apply similar fix to PFCD-2536 (wasgetting error in index creating for parition with '-' in it.
b165fea7329e5b1	2016-11-26 15:31:26	PFCD-2536	Apply similar fix to PFCD-2536 (wasgetting error in index creating for parition with '-' in it.
14ef6309fdbe6e3	2017-01-25 13:35:30	PFCD-2538	PFCD-2538 avoid NPE on invalid filter
83bef3069c0a531	2017-01-25 13:35:23	PFCD-2538	PFCD-2538 do not allow quote to be saved when there are some lineitems not matchich QPF
2fb9bdd87a7fc71	2017-01-25 13:35:15	PFCD-2538	PFCD-2538 fix npe on missing quote

a8c5a29704595fa	2017-01-25 13:35:07	PFCD-2538	PFCD-2538 reset customer on quote before QPF call
17b45d0f4ebc81f	2017-01-25 13:34:58	PFCD-2538	PFCD-2538 fix test drive for qpf
975a912cedc726e	2017-01-25 13:34:46	PFCD-2538	PFCD-2538 more fancy int test
605cc9b74a4002b	2017-01-25 13:33:40	PFCD-2538	PFCD-2538 computale quote product filter with test drive
efe4ffdbafe3bca	2017-01-25 13:30:57	PFCD-2538	PFCD-2538 add Quote Product Filter Formulas
2e3fd36e58f9f51	2017-03-31 08:20:34	PFCD-2571	PFCD-2571 check for cancelled async task not in tx
d88d3b68b6ee920	2016-12-08 15:14:15	PFCD-2571	PFCD-2571 dbmigrations server support flyway on start is disabled and hibernate is still not disabled for now PFCD-2571 cleanup a bit PFCD-2571 fogotten hunk fix inttest, schema_version is immutable and foreign table at all PFCD-2571 rename schema_version table to avoid issues with underscore PFCD-2571 table incorrectly created on oracle, renaming PFCD-2571 int_tests trigger check on SchemaVersion PFCD-2571 int_tests forgotten hunk PFCD-2571 align recommended cli with internally used in automated tests PFCD-2751 fix after rafactoring PFCD-2571 int_tests now use flyway please adjust your config also one must drop his int_tests db and create an empty one (this is needed only once after pulling this commit in) int_tests: no hacky settings neeeded anymore PFCD-2571 two location instead of pre suffix PFCD-2571 cli window shows incorrect advice PFCD-2571 logging PFCD-2571 mariadb10.1 syntax for idempotent PRE scripts move invariants to flyway and mariadb10.1 db tests: flyway schema name must be uppercase on oracle PFCD-2571 server migrations fix integration tests PFCD-2571 fail on nok response from dbmigrations repo server PFCD-2571 apache http client must be closed manually PFCD-2571 cleanup PFCD-2571 fix empty schema issue PFCD-2751 do not configure it directly PFCD-2751 fix log4jdbc url fails, since flyway is unable to pick correct driver PFCD-2571 move to hikariDS to support log4jdbc
b2dfe3768a247ea	2016-11-28 10:40:47	PFCD-2592	PFCD-2592 SandboxApi: customer/product extensions does not reflect passed filters
799c273fbed2e0d	2016-12-08 15:45:18	PFCD-2593	PFCD-2593 Open up RRs to CFSs -> implemented custom rr resolver in basesunction
7b449ac8de2bed0	2016-12-02 11:52:53	PFCD-2593	PFCD-2593 Open up RRs to CFSs -> allowing to modify rr from api
045076d82e6d54f	2016-12-02 11:08:47	PFCD-2593	PFCD-2593 Open up RRs to CFSs - oracle fix
ae4b561c60becbe	2016-12-01 18:05:37	PFCD-2593	PFCD-2593 Open up RRs to CFSs
1c9e3cf0ff90a99	2016-11-29 14:38:03	PFCD-2594	PFCD-2594 DCR no sort possible after clicking "Show Changes Only" button -> sorting was done prviously in browser, sort of works, I hav changed backed code to properly sort there
86d457759a03d27	2016-11-29 14:17:28	PFCD-2596	PFCD-2596 Catch null case if typedId is only passed in oldValues
5d050afd0b68498	2016-11-29 07:02:02	PFCD-2596	PFCD-2596 productmanager.fetch/*/PX/UnknownExtension throws NPE
cc9480bab9f498b	2016-11-30 13:24:49	PFCD-2599	PFCD-2599 Make size of configurator window formula controllable (now also from groovy)
04d4878132c68fd	2016-11-29 13:39:14	PFCD-2599	PFCD-2599 Make size of configurator window formula controllable
6d36796c896a3d4	2016-12-12 18:18:34	PFCD-2600	PFCD-2600 various PA Data Entitlements improvements - if any entitlements are defined for a DS, users with no appropriate group cannot see anything - empty filter is now a wildcard (full data acccess in DS)
09f706ed8822227	2016-11-29 15:45:30	PFCD-2602	PFCD-2602 - fixing conversion
8a3b14ecd8b90d1	2016-12-13 06:51:48	PFCD-2606	PFCD-2606 Only create zips in incremental update when element is really dirty
fcd8ff25e585a73	2016-12-01 07:47:28	PFCD-2606	PFCD-2606 Provide option for incremental compilation of formula update to speed up process
d144eb14aeafe2d	2016-12-02 07:36:59	PFCD-2609	PFCD-2609 Whitelist MLTVM for meta configurator
3a07d8c645802bc	2016-12-18 22:57:32	PFCD-2610	PFCD-2659 PFCD-2610 Sorting by Comment column in DCR do nothing Filtering of DCR in Show Changes Only do not works correctly
fbb723a126c57c5	2016-12-02 10:37:53	PFCD-2611	PFCD-2611 Properly clearing out alerts & warnings to NULL if empty (instead of "{}" (Fix test)

3032225de8cf96a	2016-12-02 09:52:56	PFCD-2611	PFCD-2611 Properly clearing out alerts & warnings to NULL if empty (instead of "{}"
4d400283447ab7e	2016-12-08 13:29:26	PFCD-2614	PFCD-2614 - added more logging
dde3cbdf054b431	2016-12-08 11:32:40	PFCD-2614	PFCD-2614 - refactor code to write tests
f4f133daac7548a	2016-12-08 07:46:00	PFCD-2632	PFCD-2632 Make SAML SSO also play with ADFS
1c8e53897ecb460	2016-12-12 16:42:15	PFCD-2647	PFCD-2647 Fix configurator with extra arguments
0a01ede06a2d969	2016-12-15 14:42:44	PFCD-2650	PFCD-2650 adding attachment unlinks lineitems and nextRevUNs
b672051534d696f	2016-12-15 06:42:04	PFCD-2654	PFCD-2654 Dynamically adding items of Matrix-LPG not working (part 2)
8f99e53e9ad5262	2016-12-14 16:44:55	PFCD-2654	PFCD-2654 Dynamically adding items of Matrix-LPG not working
c7b1fcc9616a4e1	2017-01-03 17:19:40	PFCD-2659	PFCD-2659 Sorting by Comment column in DCR do nothing -> allowing soring comments in MPL
60552050370dc82	2017-01-02 19:53:54	PFCD-2659	PFCD-2659 - fixing numeric sorting in changes mode
3a07d8c645802bc	2016-12-18 22:57:32	PFCD-2659	PFCD-2659 PFCD-2610 Sorting by Comment column in DCR do nothing Filtering of DCR in Show Changes Only do not works correctly
d2ff577bbdb3561	2016-12-17 14:47:20	PFCD-2665	fix(service) - Add chart to Dashboard failed due to NPE in FormulaManagerServiceImpl (PFCD-2665) variable pathCrumbs in this case is null
580cf34102d65f7	2016-12-21 09:29:57	PFCD-2680	PFCD-2680: PA: error loading data in some DFs
479f78b002d184a	2016-12-22 07:56:58	PFCD-2682	fix(pgi error reporting) (#PFCD-2682)
fb6a898d5e75267	2016-12-22 14:05:56	PFCD-2683	PFCD-2683: PA DL: send event on DL competion (successful or not)
e3db7a898c03964	2017-01-17 12:33:07	PFCD-2684	fix(Purge compiledClasses temp folder after saving formula) #PFCD-2684
a25305f1982353c	2017-01-18 17:14:13	PFCD-2727	PFCD-2727 -> OOM in JdbcCoordinatorImpl -> updated hibernate ot our patched version !!!! -> important for upgrade to check whether all backported issues are in newer version, probably major 5. see the issue I have put all links there
3a1eb8c867e0172	2017-01-17 13:52:25	PFCD-2727	PFCD-2727 OOM in JdbcCoordinatorImpl -> fix test
d15c41a86d10d8c	2017-01-17 12:32:06	PFCD-2727	PFCD-2727 OOM in JdbcCoordinatorImpl -> impropper formatter caused the chain reaction that with combination with hibernate bug caused OOM.
073b817c83f4a1d	2017-01-16 12:01:15	PFCD-2727	PFCD-2727 - properly set query cache config
1890416455c3c7d	2017-02-22 07:11:42	PFCD-2728	fix(dirty runs) #PFCD-2728 pass on runnumber correctly to calculation context in order to have correct api.IsSecondPass() results also in multi-dirty run configurations
45c77cec63b4a2b	2017-01-16 13:05:56	PFCD-2728	fix(dirty pass recalc) #PFCD-2728 Individual item item recalc also respects more than one dirty run now (missing import)
84754b51b852884	2017-01-16 12:58:21	PFCD-2728	fix(dirty pass recalc) #PFCD-2728 Individual item item recalc also respects more than one dirty run now
bf93431ea6af9f4	2017-01-17 13:09:36	PFCD-2732	PFCD-2732 RR email templates are missing
31152228ff2292b	2016-12-14 12:02:47	PFCD-2751	PFCD-2751 no deleted partition should be found
7a3330d70d643f0	2016-12-12 09:07:08	PFCD-2751	PFCD-2751 fix integration tests
e37a92d9916afbb	2016-12-12 08:56:51	PFCD-2751	PFCD-2751 fix test
ae6620e71df69c2	2016-12-09 14:03:41	PFCD-2751	PFCD-2751 correctly handle enabling/disabling of hibernate

26db8aa9056d867	2016-12-09 13:58:50	PFCD-2751	PFCD-2751 paging for cached results
225c715c38cc5c1	2016-12-09 13:58:37	PFCD-2751	PFCD-2751 do not print create tables for flyway table in schema check command
6b9810c87fd0cd8	2016-12-09 13:58:20	PFCD-2751	PFCD-2751 convince ORM on flyqay table to work also on oracle
4f3588ca771be17	2016-12-08 15:35:30	PFCD-2751	PFCD-2751 develop to master diff
d88d3b68b6ee920	2016-12-08 15:14:15	PFCD-2751	PFCD-2571 dbmigrations server support flyway on start is disabled and hibernate is still not disabled for now PFCD-2571 cleanup a bit PFCD-2571 fogotten hunk fix inttest, schema_version is immutable and foreign table at all PFCD-2571 rename schema_version table to avoid issues with underscore PFCD-2571 table incorrectly created on oracle, renaming PFCD-2571 int_tests trigger check on SchemaVersion PFCD-2571 int_tests forgotten hunk PFCD-2571 align recommended cli with internally used in automated tests PFCD-2751 fix after rafactoring PFCD-2571 int_tests now use flyway please adjust your config also one must drop his int_tests db and create an empty one (this is needed only once after pulling this commit in) int_tests: no hacky settings neeeded anymore PFCD-2571 two location instead of pre suffix PFCD-2571 cli window shows incorrect advice PFCD-2571 logging PFCD-2571 mariadb10.1 syntax for idempotent PRE scripts move invariants to flyway and mariadb10.1 db tests: flyway schema name must be uppercase on oracle PFCD-2571 server migrations fix integration tests PFCD-2571 fail on nok response from dbmigrations repo server PFCD-2571 apache http client must be closed manually PFCD-2571 cleanup PFCD-2571 fix empty schema issue PFCD-2751 do not configure it directly PFCD-2751 fix log4jdbc url fails, since flyway is unable to pick correct driver PFCD-2571 move to hikariDS to support log4jdbc
c99788319c104c0	2017-01-23 09:35:23	PFCD-2759	fix(sku renaming and new PX) #PFCD-2759
2b0a888da7f07ea	2017-01-31 16:11:49	PFCD-2762	PFCD-2762 - implementing on conflict strategy: validation error
c425900b5ee6f4b	2017-02-03 18:26:36	PFCD-2795	fix(PersistedObjectServiceImpl) LPG to PL converison does not work on Oracle (#PFCD-2795) back-ported to Mai Tai
97e9f7465bb1cc4	2017-02-07 14:51:45	PFCD-2805	PFCD-2805 inputs/outputs not avilable in QuoteProductFilter logic
acf5dbfebf0db30	2017-02-14 20:13:09	PFCD-2811	Alternative to commit 85e8136, which seems to cause regression issues (PFCD-2818, PFCD-2811)
610377c68ddb3db	2017-02-10 15:15:13	PFCD-2811	PFCD-2811: Rebate agreement doesn't show rebate record results in submitted state
acf5dbfebf0db30	2017-02-14 20:13:09	PFCD-2818	Alternative to commit 85e8136, which seems to cause regression issues (PFCD-2818, PFCD-2811)
9207ab8713edb7d	2017-02-14 19:47:37	PFCD-2818	fix(DefaultFormulaEngine) - reverting commit "Formula eval perf optimization: only apply 'Parameters Details' in syntax check mode" (#PFCD-2818)
83c7860781ff7d1	2017-02-24 17:17:15	PFCD-2829	fix(DMCalculation) catch IllegalFieldValuesException (#PFCD-2829) Target row update was already in a dedicated try/catch clause, but not the source row update.
a6a50f92bf07a39	2017-02-20 10:28:04	PFCD-2841	PFCD-2841: RebateRecord allocation slow due to currentItem conversions
b27507c5fbe97c4	2017-02-24 08:41:42	PFCD-2854	PFCD-2854 hql delete of DMDataChunk does not cascade to errors @ElementCollection
b4814411ae7a0d1	2017-02-28 14:47:31	PFCD-2862	[EC] Fixed DCR download issue (7751084, PFCD-2863), ProductRef upload issue with booleans (7993605, PFCD-2862), Bosch issue with large IDs (PFCD-2872)
b4814411ae7a0d1	2017-02-28 14:47:31	PFCD-2863	[EC] Fixed DCR download issue (7751084, PFCD-2863), ProductRef upload issue with booleans (7993605, PFCD-2862), Bosch issue with large IDs (PFCD-2872)
b4814411ae7a0d1	2017-02-28 14:47:31	PFCD-2872	[EC] Fixed DCR download issue (7751084, PFCD-2863), ProductRef upload issue with booleans (7993605, PFCD-2862), Bosch issue with large IDs (PFCD-2872)
b0ca40dee27327b	2017-02-28 14:03:29	PFCD-2881	fix(bulkload) #PFCD-2881 Fix loaddata for PXREF (or other types containing real booleans)
44a151a3284fefe	2017-03-01 17:23:43	PFCD-2885	fix(permissions) #PFCD-2885 CX30 is not mapped to correct permission in CRUD command
26a4334b257335a	2017-03-10 07:02:22	PFCD-2898	fix(start page) #PFCD-2898 Option to always show pending approvals and todos
5b36dd174023a49	2017-03-14 14:51:06	PFCD-2908	Revert "[EC] Fix for validation not working on drop down based cells (PFCD-2908)" This reverts commit 4a1399f4bc194b59a57342a93467239ee4f5015b.
4a1399f4bc194b5	2017-03-14 14:48:43	PFCD-2908	[EC] Fix for validation not working on drop down based cells (PFCD-2908)

62024441a97964a	2017-03-17 14:27:40	PFCD-2912	fix(PA filter query performance) PFCD-2912
e282264310b92ba	2017-03-09 10:49:14	PFCD-2919	fix(DF dataload) #PFCD-2919
b8594a602106202	2017-03-26 09:14:56	PFCD-2930	feat (docker) PFCD-2930 - add possibility for jenkins to make changes
0be65bf10b93541	2017-03-21 09:35:32	PFCD-2930	feat (docker) PFCD-2930 - ported docker file
4d010bf6f05b350	2017-03-14 09:53:43	PFCD-2934	feat(formula to XLS) #PFCD-2934 Reformat formula results and optionally use a template in case of output=XLS
264a81670210ff5	2017-03-16 09:00:59	PFCD-2939	PFCD-2939 converting nonattributed numeric fields on ORACLE in arithmetic expresions causes perfomance issue
7fee357a13f110d	2017-04-14 05:58:11	PFCD-3057	PFCD-3057 no OBJREFs to all customers for CT/RBA
a948ccfb122b592	2017-04-13 09:50:36	PFCD-3057	PFCD-3057 ensure uncached query
89297bb96213628	2017-04-13 09:50:01	PFCD-3057	PFCD-3057 postpone jst kicking after tx is commited

pricefx-server ChangeLog All changes

Commit	Date	Message
7fee357a13f110d	2017-04-14 05:58:11	PFCD-3057 no OBJREFs to all customers for CT/RBA
b7e2ec8df5486d8	2017-04-14 05:56:52	PFCD3057 avoid CT/RBA JST race condition with allowing second same OBJREF taks to be scheduled
a948ccfb122b592	2017-04-13 09:50:36	PFCD-3057 ensure uncached query
89297bb96213628	2017-04-13 09:50:01	PFCD-3057 postpone jst kicking after tx is commited
bcbe8b65493bc31	2017-04-07 08:25:41	fix(DF2DSFormulaGenerator) "Year out of range" in Oracle (#PFCD3018)
2e3fd36e58f9f51	2017-03-31 08:20:34	PFCD-2571 check for cancelled async task not in tx
b8594a602106202	2017-03-26 09:14:56	feat (docker) PFCD-2930 - add possibility for jenkins to make changes
0be65bf10b93541	2017-03-21 09:35:32	feat (docker) PFCD-2930 - ported docker file
62024441a97964a	2017-03-17 14:27:40	fix(PA filter query performance) PFCD-2912
264a81670210ff5	2017-03-16 09:00:59	PFCD-2939 converting nonattributed numeric fields on ORACLE in arithmetic expresions causes perfomance issue
6f09e2814934497	2017-03-15 15:28:37	Upping default shared cache TTL from 15 min to 2h for hazelcast instances
5b36dd174023a49	2017-03-14 14:51:06	Revert "[EC] Fix for validation not working on drop down based cells (PFCD-2908)" This reverts commit 4a1399f4bc194b59a57342a93467239ee4f5015b.
4a1399f4bc194b5	2017-03-14 14:48:43	[EC] Fix for validation not working on drop down based cells (PFCD-2908)
4e0d55fe6988234	2017-03-14 09:56:31	Missing import

4d010bf6f05b350	2017-03-14 09:53:43	feat(formula to XLS) #PFCD-2934 Reformat formula results and optionally use a template in case of output=XLS
26a4334b257335a	2017-03-10 07:02:22	fix(start page) #PFCD-2898 Option to always show pending approvals and todos
e282264310b92ba	2017-03-09 10:49:14	fix(DF dataload) #PFCD-2919
fc0933f4651074b	2017-03-01 21:46:33	[EC] Added 'comments' column to DCR object
44a151a3284fefe	2017-03-01 17:23:43	fix(permissions) #PFCD-2885 CX30 is not mapped to correct permission in CRUD command
b4814411ae7a0d1	2017-02-28 14:47:31	[EC] Fixed DCR download issue (7751084, PFCD-2863), ProductRef upload issue with booleans (7993605, PFCD-2862), Bosch issue with large IDs (PFCD-2872)
b0ca40dee27327b	2017-02-28 14:03:29	fix(bulkload) #PFCD-2881 Fix loaddata for PXREF (or other types containing real booleans)
bd7df9ffc4ee39f	2017-02-28 09:09:40	[EC] Fixed delete PX/CX for non default sized (i.e. not 30) tables
83c7860781ff7d1	2017-02-24 17:17:15	fix(DMCalculation) catch IllegalFieldValuesException (#PFCD-2829) Target row update was already in a dedicated try/catch clause, but not the source row update.
b27507c5fbe97c4	2017-02-24 08:41:42	PFCD-2854 hql delete of DMDataChunk does not cascade to errors @ElementCollection
fc0df3002177082	2017-02-22 07:53:29	Remove obsolete import
1890416455c3c7d	2017-02-22 07:11:42	fix(dirty runs) #PFCD-2728 pass on runnumber correctly to calculation context in order to have correct api.IsSecondPass() results also in multi-dirty run configurations
a6a50f92bf07a39	2017-02-20 10:28:04	PFCD-2841: RebateRecord allocation slow due to currentItem conversions
d93c574f51e560a	2017-02-17 08:13:44	Enable ADFS compliant SAML flags by default
bf5e08c777c02c3	2017-02-14 20:26:44	Missing import
acf5dbfebf0db30	2017-02-14 20:13:09	Alternative to commit 85e8136, which seems to cause regression issues (PFCD-2818, PFCD-2811)
9207ab8713edb7d	2017-02-14 19:47:37	fix(DefaultFormulaEngine) - reverting commit "Formula eval perf optimization: only apply 'Parameters Details' in syntax check mode" (#PFCD-2818)
610377c68ddb3db	2017-02-10 15:15:13	PFCD-2811: Rebate agreement doesn't show rebate record results in submitted state
97e9f7465bb1cc4	2017-02-07 14:51:45	PFCD-2805 inputs/outputs not avilable in QuoteProductFilter logic
c425900b5ee6f4b	2017-02-03 18:26:36	fix(PersistedObjectServiceImpl) LPG to PL converison does not work on Oracle (#PFCD-2795) back-ported to Mai Tai
37de616a6049441	2017-02-02 13:44:19	Remove internal field "fqloginName" from User interface definition
addf3b781210127	2017-02-02 10:45:20	Fix SessionReaper handling of session deletion (outside of loop)
8f29e9fbaf442fa	2017-02-02 07:30:22	Defaulting "rerun" to true to prevent accidental data loss. Target PX is now only cleaned if rerun = false is explicitly set
21cc9aa05dffa22	2017-01-31 18:09:00	[EC] Fix for CX and PX loaddata command to work with ALL PX sized tables

2b0a888da7f07ea	2017-01-31 16:11:49	PFCD-2762 - implementing on conflict strategy: validation error
7cfa67af031a2f5	2017-01-31 08:43:26	FlexChart 'color' attribute lost in json --> chart deserialiation
4c4a0e51ba48d15	2017-01-30 13:24:47	chore(i18n): add common "read only" text
85e8136bdbd2322	2017-01-29 20:53:42	Formula eval perf optimization: only apply 'Parameters Details' in syntax check mode
13d897b8f6824be	2017-01-25 17:02:51	fixing miss configuartion in logback
a47a5abdb62f19d	2017-01-25 14:07:42	(inttest) mising persitedObjectBuilder in master
14ef6309fdbe6e3	2017-01-25 13:35:30	PFCD-2538 avoid NPE on invalid filter
83bef3069c0a531	2017-01-25 13:35:23	PFCD-2538 do not allow quote to be saved when there are some lineitems not matchich QPF
2fb9bdd87a7fc71	2017-01-25 13:35:15	PFCD-2538 fix npe on missing quote
a8c5a29704595fa	2017-01-25 13:35:07	PFCD-2538 reset customer on quote before QPF call
17b45d0f4ebc81f	2017-01-25 13:34:58	PFCD-2538 fix test drive for qpf
975a912cedc726e	2017-01-25 13:34:46	PFCD-2538 more fancy int test
605cc9b74a4002b	2017-01-25 13:33:40	PFCD-2538 computale quote product filter with test drive
efe4ffdbafe3bca	2017-01-25 13:30:57	PFCD-2538 add Quote Product Filter Formulas
c99788319c104c0	2017-01-23 09:35:23	fix(sku renaming and new PX) #PFCD-2759
a25305f1982353c	2017-01-18 17:14:13	PFCD-2727 -> OOM in JdbcCoordinatorImpl -> updated hibernate ot our patched version !!!! -> important for upgrade to check whether all backported issues are in newer version, probably major 5. see the issue I have put all links there
3a1eb8c867e0172	2017-01-17 13:52:25	PFCD-2727 OOM in JdbcCoordinatorImpl -> fix test
bf93431ea6af9f4	2017-01-17 13:09:36	PFCD-2732 RR email templates are missing
e3db7a898c03964	2017-01-17 12:33:07	fix(Purge compiledClasses temp folder after saving formula) #PFCD-2684
d15c41a86d10d8c	2017-01-17 12:32:06	PFCD-2727 OOM in JdbcCoordinatorImpl -> impropper formatter caused the chain reaction that with combination with hibernate bug caused OOM.
45c77cec63b4a2b	2017-01-16 13:05:56	fix(dirty pass recalc) #PFCD-2728 Individual item item recalc also respects more than one dirty run now (missing import)
84754b51b852884	2017-01-16 12:58:21	fix(dirty pass recalc) #PFCD-2728 Individual item item recalc also respects more than one dirty run now
073b817c83f4a1d	2017-01-16 12:01:15	PFCD-2727 - properly set query cache config
94a23c45e547e3f	2017-01-16 11:56:07	Some extra logging around LPG calculation and 2nd passes

9dac3961cd95df6	2017-01-16 11:27:35	Some extra logging around LPG calculation and 2nd passes
b24c86e2069a9bb	2017-01-11 10:20:15	[EC] Added default sorting for PX (sku) and CX (customerId)
35ac9f8d59a6d9d	2017-01-06 08:30:28	test(PriceGridManagerService test) - to debug failing test on Oracle after Jan 1st
c2523b33675c173	2017-01-06 08:30:17	test(log message) - to debug failing test on Oracle after Jan 1st
c7b1fcc9616a4e1	2017-01-03 17:19:40	PFCD-2659 Sorting by Comment column in DCR do nothing -> allowing soring comments in MPL
60552050370dc82	2017-01-02 19:53:54	PFCD-2659 - fixing numeric sorting in changes mode
6c6f3ccf50caea9	2016-12-22 15:53:40	fix of random error
36ea9b74aadad18	2016-12-22 15:33:55	Backing out mariadb driver (as not used)
fb6a898d5e75267	2016-12-22 14:05:56	PFCD-2683: PA DL: send event on DL competion (successful or not)
479f78b002d184a	2016-12-22 07:56:58	fix(pgi error reporting) (#PFCD-2682)
65e7a2747e63da4	2016-12-21 13:07:41	Revert "PFCD-2514 allow run with older server" This reverts commit 4af0634a7aea85b22a7fefe0ea40eca6f521c60b.
4af0634a7aea85b	2016-12-21 13:04:27	PFCD-2514 allow run with older server
580cf34102d65f7	2016-12-21 09:29:57	PFCD-2680: PA: error loading data in some DFs
3a07d8c645802bc	2016-12-18 22:57:32	PFCD-2659 PFCD-2610 Sorting by Comment column in DCR do nothing Filtering of DCR in Show Changes Only do not works correctly
d2ff577bbdb3561	2016-12-17 14:47:20	fix(service) - Add chart to Dashboard failed due to NPE in FormulaManagerServiceImpl (PFCD-2665) variable pathCrumbs in this case is null
bdaac1186d03e2d	2016-12-16 08:27:37	Making maintenance mode log messges more explicit
a21c19f31c4e317	2016-12-15 16:16:23	Fix remote session determination
0a01ede06a2d969	2016-12-15 14:42:44	PFCD-2650 adding attachment unlinks lineitems and nextRevUNs
49a34cf4f3b1074	2016-12-15 13:28:45	Fix remote session determination
b672051534d696f	2016-12-15 06:42:04	PFCD-2654 Dynamically adding items of Matrix-LPG not working (part 2)
8f99e53e9ad5262	2016-12-14 16:44:55	PFCD-2654 Dynamically adding items of Matrix-LPG not working
f091c4de19dfc03	2016-12-14 12:44:04	PFCD-2514 pdf export button available only when clouconvert enabled and xls template present
31152228ff2292b	2016-12-14 12:02:47	PFCD-2751 no deleted partition should be found
ba05e9f148ca278	2016-12-14 10:03:00	Adding more logging to XPGI matrix formula execution

4b0b0b13f0f245d	2016-12-13 15:15:12	PFCD-2519 PA calculations: formula errors not resulting in ERROR status
05d0ce157794404	2016-12-13 11:02:16	Add mariadb jdbc driver as optionally configurable package Optimize remote session determination and omit duplicates
8a6dda1c7776222	2016-12-13 07:49:23	Disabling compression on event message
8a3b14ecd8b90d1	2016-12-13 06:51:48	PFCD-2606 Only create zips in incremental update when element is really dirty
ee5b7e86218e186	2016-12-12 19:04:15	Ensure timely delivery of events & emails also when system is low-loaded and has many remote schemas
3a9c097c880b259	2016-12-12 18:50:19	Adding more loggin to EventTask processor
6d36796c896a3d4	2016-12-12 18:18:34	PFCD-2600 various PA Data Entitlements improvements - if any entitlements are defined for a DS, users with no appropriate group cannot see anything - empty filter is now a wildcard (full data acccess in DS)
1c8e53897ecb460	2016-12-12 16:42:15	PFCD-2647 Fix configurator with extra arguments
7a3330d70d643f0	2016-12-12 09:07:08	PFCD-2751 fix integration tests
e37a92d9916afbb	2016-12-12 08:56:51	PFCD-2751 fix test
ae6620e71df69c2	2016-12-09 14:03:41	PFCD-2751 correctly handle enabling/disabling of hibernate
26db8aa9056d867	2016-12-09 13:58:50	PFCD-2751 paging for cached results
225c715c38cc5c1	2016-12-09 13:58:37	PFCD-2751 do not print create tables for flyway table in schema check command
6b9810c87fd0cd8	2016-12-09 13:58:20	PFCD-2751 convince ORM on flyqay table to work also on oracle
8c92bea840afd18	2016-12-08 17:05:06	temporarily disable hibernate
799c273fbed2e0d	2016-12-08 15:45:18	PFCD-2593 Open up RRs to CFSs -> implemented custom rr resolver in basesunction
4f3588ca771be17	2016-12-08 15:35:30	PFCD-2751 develop to master diff
d88d3b68b6ee920	2016-12-08 15:14:15	PFCD-2571 dbmigrations server support flyway on start is disabled and hibernate is still not disabled for now PFCD-2571 cleanup a bit PFCD-2571 fogotten hunk fix inttest, schema_version is immutable and foreign table at all PFCD-2571 rename schema_version table to avoid issues with underscore PFCD-2571 table incorrectly created on oracle, renaming PFCD-2571 int_tests trigger check on SchemaVersion PFCD-2571 int_tests forgotten hunk PFCD-2571 align recommended cli with internally used in automated tests PFCD-2751 fix after rafactoring PFCD-2571 int_tests now use flyway please adjust your config also one must drop his int_tests db and create an empty one (this is needed only once after pulling this commit in) int_tests: no hacky settings neeeded anymore PFCD-2571 two location instead of pre suffix PFCD-2571 cli window shows incorrect advice PFCD-2571 logging PFCD-2571 mariadb10.1 syntax for idempotent PRE scripts move invariants to flyway and mariadb10.1 db tests: flyway schema name must be uppercase on oracle PFCD-2571 server migrations fix integration tests PFCD-2571 fail on nok response from dbmigrations repo server PFCD-2571 apache http client must be closed manually PFCD-2571 cleanup PFCD-2571 fix empty schema issue PFCD-2751 do not configure it directly PFCD-2751 fix log4jdbc url fails, since flyway is unable to pick correct driver PFCD-2571 move to hikariDS to support log4jdbc
37057df60fe136c	2016-12-08 13:45:58	Removing logstash log delivery from "performance" category
4d400283447ab7e	2016-12-08 13:29:26	PFCD-2614 - added more logging

dde3cbdf054b431	2016-12-08 11:32:40	PFCD-2614 - refactor code to write tests
f4f133daac7548a	2016-12-08 07:46:00	PFCD-2632 Make SAML SSO also play with ADFS
e04ad37d486feb2	2016-12-07 17:44:22	PFCD-2527 PA: TOSTRING expression was not returning String type
b896da3105ee050	2016-12-07 10:02:10	Minor i18n fixes
817ecc1ecb7cb3f	2016-12-02 12:36:12	no customerextension and productextension table in MaiTai, hence no triggers needed
7b449ac8de2bed0	2016-12-02 11:52:53	PFCD-2593 Open up RRs to CFSs -> allowing to modify rr from api
045076d82e6d54f	2016-12-02 11:08:47	PFCD-2593 Open up RRs to CFSs - oracle fix
fbb723a126c57c5	2016-12-02 10:37:53	PFCD-2611 Properly clearing out alerts & warnings to NULL if empty (instead of "{}" (Fix test)
3032225de8cf96a	2016-12-02 09:52:56	PFCD-2611 Properly clearing out alerts & warnings to NULL if empty (instead of "{}"
d144eb14aeafe2d	2016-12-02 07:36:59	PFCD-2609 Whitelist MLTVM for meta configurator
ae4b561c60becbe	2016-12-01 18:05:37	PFCD-2593 Open up RRs to CFSs
c53d68ec9c57dd7	2016-12-01 14:17:18	PFCD-2282 Handle errors in the feeder logic of PA Calculation DLs Both formula errors and any other feeder exception were ignored. Also added JST progress info updates.
fcd8ff25e585a73	2016-12-01 07:47:28	PFCD-2606 Provide option for incremental compilation of formula update to speed up process
cc9480bab9f498b	2016-11-30 13:24:49	PFCD-2599 Make size of configurator window formula controllable (now also from groovy)
09f706ed8822227	2016-11-29 15:45:30	PFCD-2602 - fixing conversion
1c9e3cf0ff90a99	2016-11-29 14:38:03	PFCD-2594 DCR no sort possible after clicking "Show Changes Only" button -> sorting was done prviously in browser, sort of works, I hav changed backed code to properly sort there
86d457759a03d27	2016-11-29 14:17:28	PFCD-2596 Catch null case if typedId is only passed in oldValues
04d4878132c68fd	2016-11-29 13:39:14	PFCD-2599 Make size of configurator window formula controllable
64e1e1e53110f8b	2016-11-29 09:42:08	PFCD-2205 clouconvert agreement token can't be read correctly on non-remote schemas
5d050afd0b68498	2016-11-29 07:02:02	PFCD-2596 productmanager.fetch/*/PX/UnknownExtension throws NPE
b2dfe3768a247ea	2016-11-28 10:40:47	PFCD-2592 SandboxApi: customer/product extensions does not reflect passed filters
98251c8907951e7	2016-11-26 15:36:47	Apply similar fix to PFCD-2536 (wasgetting error in index creating for parition with '-' in it.
766d030aae49959	2016-11-26 15:35:27	Pulling latest master, which should be same as 3.1 at release time
b165fea7329e5b1	2016-11-26 15:31:26	Apply similar fix to PFCD-2536 (wasgetting error in index creating for parition with '-' in it.

3972231c6b51060	2016-11-26 07:48:17	Added CloudConvert API token step
5b596ce84edd471	2016-11-25 20:12:11	chore(i18N): add deny other reason hint eng text

Mai Tai 3.1 pricefx-web Changelog

- pricefx-web ChangeLog Issue Specific
- pricefx-web ChangeLog All changes

pricefx-web ChangeLog Issue Specific

Commit	Date	Issue	Message
10d4ed6e61c0d46	2017-01-04 11:32:46	PFCD-1322	feat(AdvancedGrid): improve column reorder (#PFCD-1322)
67884d8fa377219	2016-11-26 19:32:05	PFCD-2205	PFCD-2205 cloudconvert system config does not work on virtual system partition
e8e9cbf77f1edfe	2017-03-14 15:53:49	PFCD-2214	fix (mass edit on negative value) PFCD-2214 - also integer input supported
9f4ceafe21e09cf	2017-03-14 15:53:23	PFCD-2214	fix (mass edit for negative numbers) PFCD-2214 : Mass Edit does not allow change to "-1", decimal values only
52e7ead50b972f1	2016-12-12 16:43:20	PFCD-2250	PFCD-2250 minor label fix
f1e5bf2a3440cf5	2017-01-17 13:13:46	PFCD-2308	PFCD-2308 Unable to open/approve RR from My Pending Approvals dashlet
379b2c6fdc0ed96	2016-11-30 17:10:24	PFCD-2419	PFCD-2419 - LPG Detail View/Portlet View - All columns editable
261bc4e64843368	2016-11-29 14:23:48	PFCD-2452	PFCD-2452 Show also individual alerts in a way that it is not overlayed by a critical alert
b91d9137ba52837	2016-12-21 13:09:08	PFCD-2514	PFCD-2514 allow run with older server
da056e3f0f277af	2016-12-14 12:45:18	PFCD-2514	PFCD-2514 pdf export button available only when clouconvert enabled and xls template present
abfc03f578e5f2e	2017-01-25 13:47:46	PFCD-2538	PFCD-2538 tiny cleanup
1e192dcf71ab689	2017-01-25 13:47:25	PFCD-2538	PFCD-2538 just set quote dto before product pikcer open, do not rebuild ui
9d1d06bc749558e	2017-01-25 13:47:17	PFCD-2538	PFCD-2538 cleanup
bbee9cdde1ae060	2017-01-25 13:46:55	PFCD-2538	PFCD-2538 quote product filter formulas ui
d7c3df2730c7ef9	2017-01-25 13:45:55	PFCD-2538	PFCD-2538 initial Quote Product Filter formulas
3d4d59cd8eb7bf7	2016-12-06 15:17:16	PFCD-2556	PFCD-2556 Small array initialization fix to avoid an ugly stack trace
ca89af767ca4f06	2016-12-08 15:14:48	PFCD-2571	PFCD-2571 dbmigrations ui PFCD-2571 cleanup a bit PFCD-2571 two locations instaed of sql suffix for PREs PFCD-2571 rename Baseline button to Baseline V1 PFCD-2571 server migrations PFCD-2571 cleanup
0835cea6683c89e	2016-12-13 16:05:26	PFCD-2587	Revert "PFCD-2587 Disabling event message filtering" This reverts commit a24b6664c15cea766a0002b9ad288f967520bf3a.
a24b6664c15cea7	2016-11-28 15:44:49	PFCD-2587	PFCD-2587 Disabling event message filtering
5a7b262b4080f2b	2016-12-07 23:34:30	PFCD-2590	PFCD-2590 PFCD-2628 (aggregation selection from ListGridFields, i.e. chart controller and sim query editor)
f82ed55b30b275b	2016-12-01 15:44:34	PFCD-2590	merge fixe in prevoius PFCD-2590 commit

3618afcba08eb9a	2016-12-01 15:26:29	PFCD-2590	merge fix in PFCD-2590
5af74b3bf482b0d	2016-12-01 15:18:59	PFCD-2590	Missing import in previous PFCD-2590 commit
3898909886c9394	2016-11-29 14:18:37	PFCD-2590	PFCD-2590: Dimensions vs measures aggregation functions in Additional Measures
aba364193fbb178	2016-12-02 10:28:13	PFCD-2593	PFCD-2593 Open up RRs to CFSs
c21910f617e921d	2016-11-29 09:35:26	PFCD-2597	PFCD-2597 PX/CX admin grid was not editable (label, user groups,...)
ae3b143937a8f39	2016-11-29 13:35:36	PFCD-2599	PFCD-2599 Make size of configurator window formula controllable
ee2843ef04237d8	2016-12-12 18:18:38	PFCD-2600	PFCD-2600 various PA Data Entitlements improvements - if any entitlements are defined for a DS, users with no appropriate group cannot see anything - empty filter is now a wildcard (full data acccess in DS) - allow deleting multiple rows in entitlements editor
1c9906c0d18c4be	2016-12-01 14:08:29	PFCD-2603	fix(grids): preserve the selection when editing a cell (#PFCD-2603)
9ebc6a2929f9840	2016-12-17 22:05:23	PFCD-2613	PFCD-2613 - show changes are now remebered. Last choosen value is used
f0d55f80b0724b5	2016-12-14 16:08:11	PFCD-2613	PFCD-2625 - fix add button PFCD-2640 - fix scrolling PFCD-2613 - refactoring
f2a14cac6fd6780	2017-01-02 16:56:23	PFCD-2615	refactor(combination types icons): new colors (#PFCD-2615)
f1237172783afad	2016-12-05 14:11:20	PFCD-2617	PFCD-2617 Copy action not working in PA Simulations grid
604b46a00fe8e9c	2016-12-20 09:15:43	PFCD-2619	PFCD-2619: value from expression editor was not stored
a88cd67e7f24caa	2016-12-05 18:59:50	PFCD-2619	PFCD-2619 Charts issue: wrong aggr. expression in additional measures Experimental, but no danger to deploy it
1e8f7623f505d69	2016-12-06 07:30:22	PFCD-2620	PFCD-2620 PFCD-2621 Fix message templates page
1e8f7623f505d69	2016-12-06 07:30:22	PFCD-2621	PFCD-2620 PFCD-2621 Fix message templates page
f0d55f80b0724b5	2016-12-14 16:08:11	PFCD-2625	PFCD-2625 - fix add button PFCD-2640 - fix scrolling PFCD-2613 - refactoring
5a7b262b4080f2b	2016-12-07 23:34:30	PFCD-2628	PFCD-2590 PFCD-2628 (aggregation selection from ListGridFields, i.e. chart controller and sim query editor)
2138af9428f26f5	2016-12-07 12:03:03	PFCD-2629	fix(column reorder): enable multi selection via checkboxes (#PFCD-2629)
78c50ec2318936f	2016-12-07 13:36:05	PFCD-2631	PFCD-2631 Show duration column in JST
59d4d7a7f1a4778	2016-12-08 14:53:55	PFCD-2633	fix(Contracts screen): Contracts: Auto refresh after go back to list page (#PFCD-2633)
197a9b2efda73b3	2016-12-08 18:59:01	PFCD-2635	PFCD-2635: Dimension filters fail on Dashboard:; in-use DS was being destroyed in DS cache mgmt
f0d55f80b0724b5	2016-12-14 16:08:11	PFCD-2640	PFCD-2625 - fix add button PFCD-2640 - fix scrolling PFCD-2613 - refactoring
d2a96d93da644c3	2016-12-22 19:44:52	PFCD-2641	fix(page) - Opening ConfigurationPage twice causes UI to hang on "Loading data..." modal dialog (#PFCD-2641)
c00c3dc6c35dc13	2017-01-10 18:08:20	PFCD-2659	PFCD-2659 Sorting by Comment column in DCR do nothing -> remove all comments impropper filtering from filter
2bf3531ecaf816d	2017-01-03 17:18:00	PFCD-2659	PFCD-2659 Sorting by Comment column in DCR do nothing -> sorting by MPL comments is allowed
6e0bbd8a6e5741c	2017-01-02 19:55:05	PFCD-2659	PFCD-2659 - explicit disable order by comments in show all mode
644c8854957a8dc	2017-01-23 15:02:20	PFCD-2660	PFCD-2660 - minnior fixes to reduce reflow

58f858f7263b4ad	2016-12-21 16:57:11	PFCD-2662	fix(main nav): fix ms edge layout issue (#PFCD-2662)
7d8a480bdeb0679	2016-12-19 09:57:03	PFCD-2663	PFCD-2663: Data Analyzer: Cannot change type of additional measures
62c78992d4c4d83	2017-01-02 10:44:45	PFCD-2664	fix(AdvancedListGrid): fix bottom-top range selection (#PFCD-2664)
3db48ace05cfbfb	2016-12-15 15:39:24	PFCD-2664	fix(grids): fix range selection and improve responsiveness (#PFCD-2664)
ab540ecf992ab6b	2016-12-20 13:55:42	PFCD-2666	fix(JSConversionUtil.convertToDate): skip also function objects (#PFCD-2666)
db82632d7247a6d	2017-01-04 13:43:23	PFCD-2667	PFCD-2667 - autofit is now preserved, and also prefferences are applied
c0fc4e73a9bd08e	2016-12-18 15:40:04	PFCD-2667	PFCD-2667 Auto-fit, column order and sorting are reset after Show Changes Only is clicked in a DCR -> fixed by propper reordering
09d41d891a73cd4	2016-12-22 08:34:57	PFCD-2668	fix(PL config grid) #PFCD-2668 Make output grid scrollable in PL config view
9e6e73ec77e431c	2017-03-14 09:59:49	PFCD-2669	fix (disable submit on validfation error) PFCD-2669: minnor fixes
60091dfd5667f58	2017-03-13 21:05:00	PFCD-2669	fix (disable submit on validfation error) PFCD-2669: connecting to model instead of custom handlers on components -> result is now stable. Model has to be saved at leat for folder types of line items. Implemented validation in Dao
70745efaf7394ad	2017-03-07 15:19:49	PFCD-2669	fix (validation in calculable editor) PFCD-2669 - added editor validation event for evenry imput of lineitem
45d01534ee1103c	2017-03-07 15:15:11	PFCD-2669	Revert "fix (validation in calculable editor) PFCD-2669 - added editor validation event for evenry imput of lineitem" This reverts commit a427afaccd083bd808f7ac005d6f03d9bf5b6df5.
a427afaccd083bd	2017-03-07 14:21:51	PFCD-2669	fix (validation in calculable editor) PFCD-2669 - added editor validation event for evenry imput of lineitem
9ff11b970fa6ccd	2017-01-07 13:04:57	PFCD-2669	PFCD-2669 fix -> handling also errors ina logic -> small refactor
b2d82c7b4a05474	2017-01-07 12:02:53	PFCD-2669	PFCD-2669 fix - possible null pointer
ff1556c1e49d44d	2017-01-05 16:35:36	PFCD-2669	PFCD-2669 It is possible to submit Contract with calculation errors and empty Required input fields -> checking for empty required fields in lineitems -> checking for line item resolution status -> checking in every lineitem editor
3234fda379e568b	2016-12-20 12:09:17	PFCD-2671	fix(AdvancedListGrid): workaround to prevent the browser freezing when the grouping is enabled (#PFCD-2671)
7e2dde68f4b73f6	2017-02-16 20:28:07	PFCD-2673	fix(PP): make matrix table not editable editable when the user is not entitled to edit (#PFCD-2673)
0aa7c795eb1f985	2017-02-16 13:39:33	PFCD-2673	fix(PP): hide copy and delete buttons when a pp matrix is selected and the user is not entitled to edit (#PFCD-2673)
f728d0e472a3a84	2017-02-13 13:04:57	PFCD-2673	fix(PP): set PP matrix grid not editable if the user is not entitled to edit (#PFCD-2673)
94ac9a194db8077	2017-01-30 13:24:05	PFCD-2673	refactor(PP): reuse existing user group checking method and add an info text if the PP values grid is not editable (#PFCD-2673)
0e332f91d2bab71	2017-01-30 12:38:53	PFCD-2673	fix(PP): make the PP values grid not editable if the user is not entitle (#PFCD-2673)
335e84ab826afd6	2017-01-06 13:33:34	PFCD-2679	feat(ProductDetailsEditorPortalView): refresh product details list when a item is approved or rejected (#PFCD-2679 / #PFCD-2708)
3001726e74d9aa8	2017-01-04 16:25:31	PFCD-2679	feat(AdvancedListGrid): add option to preserve the current selection between refresh (#PFCD-2679 / #PFCD-2702)
feca12e26508074	2017-01-04 13:55:25	PFCD-2679	feat(ProductDetailsEditorPortalView): scroll to the selected product on opening (#PFCD-2679 / #PFCD-2701)
15cf0a29f7bde05	2017-01-03 14:05:39	PFCD-2681	change(drag select in pde view) #PFCD-2681
acadfba6f1d601d	2017-01-02 12:52:26	PFCD-2685	PFCD-2685 After switch between views in DCR approve/deny buttons disappear

821f5ce3eddc1c2	2017-02-13 14:27:43	PFCD-2694	fix(PP): refresh the grid after copy (#PFCD-2694)
e8d3270bdc2071e	2017-01-31 10:25:24	PFCD-2694	fix(PP): invalidate the ds values cache after copying a table (#PFCD-2694)
feca12e26508074	2017-01-04 13:55:25	PFCD-2701	feat(ProductDetailsEditorPortalView): scroll to the selected product on opening (#PFCD-2679 / #PFCD-2701)
bf0e556e16eac87	2017-01-26 12:57:57	PFCD-2702	feat(LPG/PGI details): preserve the selection between LPG and PGI details (#PFCD-2702)
3001726e74d9aa8	2017-01-04 16:25:31	PFCD-2702	feat(AdvancedListGrid): add option to preserve the current selection between refresh (#PFCD-2679 / #PFCD-2702)
da092f1af32697b	2017-01-09 10:21:18	PFCD-2703	fix(pfx theme): improve text editor style (#PFCD-2703)
eed2ee558be3f03	2017-01-27 13:58:46	PFCD-2708	fix(LPG/PGI details): move to the next product only after the user complete the interaction with it (#PFCD-2708)
335e84ab826afd6	2017-01-06 13:33:34	PFCD-2708	feat(ProductDetailsEditorPortalView): refresh product details list when a item is approved or rejected (#PFCD-2679 / #PFCD-2708)
6dedf7baee0e8b3	2017-01-31 10:52:13	PFCD-2715	fix(login): auto focus username (#PFCD-2715)
a657eb948b42dc2	2017-02-22 14:27:49	PFCD-2716	feat(AdvancedListGrid): add simple selection type for cell selection (#PFCD-2716)
3d591cfa6891328	2017-01-17 13:13:30	PFCD-2732	PFCD-2732 RR email templates are missing
13b106311270f52	2017-01-19 17:38:21	PFCD-2739	fix(FieldPicker) PASimulations picker missing some fields #PFCD-2739 When a field picker's source is a DM or Sim DM only underlying DS fields were listed. Now all numeric or dimension fields of the DM are there. This change also affects the query editor Customer, Product and Rebate Queries in the Configurations menu.
d9ef4905b5230be	2017-02-01 09:23:37	PFCD-2758	fix(PA Simulations) validation errors (#PFCD-2758)
0d2cacc503fe2cb	2017-01-25 10:19:40	PFCD-2761	perf(PA FieldCollectionEditor): when switching collections (#PFCD-2761) Switching Datasources/Datamarts/Simulations was really slow, now it improved to somewhat slow.
722aadf4455295f	2017-02-03 14:02:01	PFCD-2767	fix(DCR) - Mark/Copy field value do not works in DCR views (#PFCD-2767)
b1410c3e894ed01	2017-02-03 14:01:52	PFCD-2767	fix(PL) - Mark/Copy field value do not works in DCR views (#PFCD-2767)
a3a96665bd16403	2017-02-24 15:49:46	PFCD-2769	fix(PP): init properly the grid group fields (#PFCD-2769)
f487a350fb38f84	2017-02-01 09:23:38	PFCD-2773	fix(FieldPicker) avoid SGWT warning in the console (#PFCD-2773) warning was wrong, but it makes sense to not make the offending call anyway
1a2bf62da109993	2017-02-13 12:52:45	PFCD-2775	fix(PP): hide all the editable action buttons if the user is not entitled to modify the values (#PFCD-2775)
96603f8c8499c16	2017-01-30 14:28:51	PFCD-2775	fic(PP): hide editable buttons if the user is not entitled to modify the values (#PFCD-2775)
31f9c8d3bfda8da	2017-01-31 15:18:14	PFCD-2776	fix(PP): fix css for normal dialog (#PFCD-2776)
da00428793be68e	2017-01-31 09:07:15	PFCD-2779	fix(missing required fields in CFS mapping) #PFCD-2779
c8f0ce751f7a70a	2017-02-03 07:51:18	PFCD-2785	feature(PX,CX) - PX - add SKU as busines key (#PFCD-2785)
984f8129024de24	2017-02-02 10:14:40	PFCD-2789	fix(PP): It's not possible to open MATRIX PPs (#PFCD-2789)
175e240f75b4ab7	2017-02-10 11:13:54	PFCD-2810	fix(agb link) #PFCD-2810
3b0e7f87800915a	2017-02-15 09:29:30	PFCD-2813	PFCD-2813 - add missing new disabled icon
434a1992a84215a	2017-02-14 23:57:58	PFCD-2813	PFCD-2813 Input matrix does not work well with rebate records - fixing data setup
c0477bd65c17b46	2017-02-24 13:25:03	PFCD-2817	fix(LPD): hide the unused context menu option "Filter Using" (#PFCD-2817)

318b8077bb46a21	2017-02-28 17:47:56	PFCD-2822	fix(input parameters): fix required value alert and allow to set an empty value for required input (#PFCD-2822)
4aef62bfba66121	2017-02-16 14:15:18	PFCD-2824	fix(PP): make table editable if the user group edit is null (#PFCD-2824)
d6517196e516cf1	2017-02-21 09:58:28	PFCD-2826	fix(PP): fix "copy" button visibility in values grids (#PFCD-2826)
8dda4919ff05d5c	2017-02-20 12:26:43	PFCD-2826	fix(PP): hide not used "edit" button in values grids (#PFCD-2826)
25937866776d7ba	2017-02-20 12:16:21	PFCD-2826	fix(PP): hide not used "mass submit" button (#PFCD-2826)
3d48671f37f56bf	2017-02-20 08:47:02	PFCD-2826	fix(PP): hide "change history" button if user not entitled for not matrix pp (#PFCD-2826)
95d1c1afb50ae67	2017-02-22 15:39:53	PFCD-2842	fix(AdvancedListGrid): scroll to first selected item even if it was in cache (#PFCD-2842)
c592f46af95ff9a	2017-02-22 12:21:34	PFCD-2842	fix(AdvancedGrid): re-select all the stored records (#PFCD-2842)
d48bd22c764b881	2017-02-21 15:07:18	PFCD-2842	fix(lpg/pgi): select the right record(s) even if it is not present in the grid's cache (#PFCD-2842)
9486357b49b15c6	2017-02-21 14:08:54	PFCD-2843	fix(PASims) unable to mark text field as dimension (#PFCD-2843) The Dimension column of the field collection editor was totally disabled for PA Simulations. We could not come up with a good reason to lock this out, so now it's enabled. Not just for calculated fields, but for any field.
d4c2ca65adef2a4	2017-02-24 12:38:27	PFCD-2850	fix(RA/RR): remove the showFilterBuilder from the view constructors (#PFCD-2850)
dfc7a59a0057ab4	2017-03-02 16:35:29	PFCD-2862	[EC] Fix for add PXREF (PFCD-2862)
01e9b8a35dafd5b	2017-02-28 14:47:48	PFCD-2862	[EC] Fixed DCR download issue (7751084, PFCD-2863), ProductRef upload issue with booleans (7993605, PFCD-2862), Bosch issue with large IDs (PFCD-2872)
01e9b8a35dafd5b	2017-02-28 14:47:48	PFCD-2863	[EC] Fixed DCR download issue (7751084, PFCD-2863), ProductRef upload issue with booleans (7993605, PFCD-2862), Bosch issue with large IDs (PFCD-2872)
4a8db86e478339e	2017-03-01 08:25:53	PFCD-2869	PFCD-2869 fix blowup when having attrmeta with id > Integer.MAX_VALUE
6a8bd9be93ebc2b	2017-02-28 09:16:58	PFCD-2870	fix(LPG/PGI): move to the next item in the nav grid after approve/deny (#PFCD-2870)
01e9b8a35dafd5b	2017-02-28 14:47:48	PFCD-2872	[EC] Fixed DCR download issue (7751084, PFCD-2863), ProductRef upload issue with booleans (7993605, PFCD-2862), Bosch issue with large IDs (PFCD-2872)
ca3f343f2c4d975	2017-03-13 10:00:18	PFCD-2916	chore(): add some extra logging to better investigate the issue #PFCD-2916
428e921e76733c4	2017-03-10 06:47:22	PFCD-2920	fix(users) #PFCD-2920 Allow new longer TLDs
1515bc332d5185f	2017-03-23 08:33:39	PFCD-2955	fix (create change request button) PFCD-2955 -> fixed issue with reseting model properly on every user action (add tab, close tab, on tab click)

pricefx-web ChangeLog All changes

Commit	Date	Message
1515bc332d5185f	2017-03-23 08:33:39	fix (create change request button) PFCD-2955 -> fixed issue with reseting model properly on every user action (add tab, close tab, on tab click)
50269348cc4ac67	2017-03-17 13:51:00	fix(PP): force refresh of the nav grid after approve/deny
e8e9cbf77f1edfe	2017-03-14 15:53:49	fix (mass edit on negative value) PFCD-2214 - also integer input supported
9f4ceafe21e09cf	2017-03-14 15:53:23	fix (mass edit for negative numbers) PFCD-2214 : Mass Edit does not allow change to "-1", decimal values only

9e6e73ec77e431c	2017-03-14 09:59:49	fix (disable submit on validfation error) PFCD-2669: minnor fixes
60091dfd5667f58	2017-03-13 21:05:00	fix (disable submit on validfation error) PFCD-2669: connecting to model instead of custom handlers on components -> result is now stable. Model has to be saved at leat for folder types of line items. Implemented validation in Dao
ca3f343f2c4d975	2017-03-13 10:00:18	chore(): add some extra logging to better investigate the issue #PFCD-2916
428e921e76733c4	2017-03-10 06:47:22	fix(users) #PFCD-2920 Allow new longer TLDs
70745efaf7394ad	2017-03-07 15:19:49	fix (validation in calculable editor) PFCD-2669 - added editor validation event for evenry imput of lineitem
45d01534ee1103c	2017-03-07 15:15:11	Revert "fix (validation in calculable editor) PFCD-2669 - added editor validation event for evenry imput of lineitem" This reverts commit a427afaccd083bd808f7ac005d6f03d9bf5b6df5.
a427afaccd083bd	2017-03-07 14:21:51	fix (validation in calculable editor) PFCD-2669 - added editor validation event for evenry imput of lineitem
dfc7a59a0057ab4	2017-03-02 16:35:29	[EC] Fix for add PXREF (PFCD-2862)
4a8db86e478339e	2017-03-01 08:25:53	PFCD-2869 fix blowup when having attrmeta with id > Integer.MAX_VALUE
318b8077bb46a21	2017-02-28 17:47:56	fix(input parameters): fix required value alert and allow to set an empty value for required input (#PFCD-2822)
0961068d74c6cbc	2017-02-28 16:35:19	style(): code clean up
01e9b8a35dafd5b	2017-02-28 14:47:48	[EC] Fixed DCR download issue (7751084, PFCD-2863), ProductRef upload issue with booleans (7993605, PFCD-2862), Bosch issue with large IDs (PFCD-2872)
6a8bd9be93ebc2b	2017-02-28 09:16:58	fix(LPG/PGI): move to the next item in the nav grid after approve/deny (#PFCD-2870)
dc69cb82926a1a4	2017-02-28 09:09:59	[EC] Fixed delete PX/CX for non default sized (i.e. not 30) tables
aa5d51ebb8f03e1	2017-02-24 15:55:34	Revert "chore(): fix spaces (?)" This reverts commit f8067a593e20eb22b2de23c8460d378d707faeb8.
f8067a593e20eb2	2017-02-24 15:53:46	chore(): fix spaces (?)
a3a96665bd16403	2017-02-24 15:49:46	fix(PP): init properly the grid group fields (#PFCD-2769)
c0477bd65c17b46	2017-02-24 13:25:03	fix(LPD): hide the unused context menu option "Filter Using" (#PFCD-2817)
d4c2ca65adef2a4	2017-02-24 12:38:27	fix(RA/RR): remove the showFilterBuilder from the view constructors (#PFCD-2850)
95d1c1afb50ae67	2017-02-22 15:39:53	fix(AdvancedListGrid): scroll to first selected item even if it was in cache (#PFCD-2842)
a657eb948b42dc2	2017-02-22 14:27:49	feat(AdvancedListGrid): add simple selection type for cell selection (#PFCD-2716)
c592f46af95ff9a	2017-02-22 12:21:34	fix(AdvancedGrid): re-select all the stored records (#PFCD-2842)
d48bd22c764b881	2017-02-21 15:07:18	fix(lpg/pgi): select the right record(s) even if it is not present in the grid's cache (#PFCD-2842)
9486357b49b15c6	2017-02-21 14:08:54	fix(PASims) unable to mark text field as dimension (#PFCD-2843) The Dimension column of the field collection editor was totally disabled for PA Simulations. We could not come up with a good reason to lock this out, so now it's enabled. Not just for calculated fields, but for any field.

d6517196e516cf1	2017-02-21 09:58:28	fix(PP): fix "copy" button visibility in values grids (#PFCD-2826)
8dda4919ff05d5c	2017-02-20 12:26:43	fix(PP): hide not used "edit" button in values grids (#PFCD-2826)
25937866776d7ba	2017-02-20 12:16:21	fix(PP): hide not used "mass submit" button (#PFCD-2826)
4a4452602688bdb	2017-02-20 11:45:00	refactor(PP): init the lookup values grid like the matrix
3d48671f37f56bf	2017-02-20 08:47:02	fix(PP): hide "change history" button if user not entitled for not matrix pp (#PFCD-2826)
7e2dde68f4b73f6	2017-02-16 20:28:07	fix(PP): make matrix table not editable editable when the user is not entitled to edit (#PFCD-2673)
4aef62bfba66121	2017-02-16 14:15:18	fix(PP): make table editable if the user group edit is null (#PFCD-2824)
0aa7c795eb1f985	2017-02-16 13:39:33	fix(PP): hide copy and delete buttons when a pp matrix is selected and the user is not entitled to edit (#PFCD-2673)
3b0e7f87800915a	2017-02-15 09:29:30	PFCD-2813 - add missing new disabled icon
434a1992a84215a	2017-02-14 23:57:58	PFCD-2813 Input matrix does not work well with rebate records - fixing data setup
3e31161e96e3021	2017-02-13 14:29:01	refactor(PP): use the AdvancedListGrid.setCanEdit to disable the value grids when the user are not entitled to edit
1a7675c2dab6342	2017-02-13 14:28:31	fix(PP): add matrixValuesGrid presence checking on delete row handler to prevent NPE
821f5ce3eddc1c2	2017-02-13 14:27:43	fix(PP): refresh the grid after copy (#PFCD-2694)
f728d0e472a3a84	2017-02-13 13:04:57	fix(PP): set PP matrix grid not editable if the user is not entitled to edit (#PFCD-2673)
1a2bf62da109993	2017-02-13 12:52:45	fix(PP): hide all the editable action buttons if the user is not entitled to modify the values (#PFCD-2775)
175e240f75b4ab7	2017-02-10 11:13:54	fix(agb link) #PFCD-2810
6c063e7c3cbfc78	2017-02-09 08:35:07	[EC] Fix for CX and PX loaddata command to work with ALL PX sized tables. Fixed incomplete CX portion.
63bc83f3276e2f6	2017-02-07 14:29:49	[EC] Fix for handling of PX tables with non-detailed meta information (i.e. no entry for 'numberOfItems') - Support tickets: 7897815, 7857500
722aadf4455295f	2017-02-03 14:02:01	fix(DCR) - Mark/Copy field value do not works in DCR views (#PFCD-2767)
b1410c3e894ed01	2017-02-03 14:01:52	fix(PL) - Mark/Copy field value do not works in DCR views (#PFCD-2767)
c8f0ce751f7a70a	2017-02-03 07:51:18	feature(PX,CX) - PX - add SKU as busines key (#PFCD-2785)
984f8129024de24	2017-02-02 10:14:40	fix(PP): It's not possible to open MATRIX PPs (#PFCD-2789)
36ec0b0f2e9357a	2017-02-02 09:29:12	fix(PP): fix matrix exception
f487a350fb38f84	2017-02-01 09:23:38	fix(FieldPicker) avoid SGWT warning in the console (#PFCD-2773) warning was wrong, but it makes sense to not make the offending call anyway

9fb9af232633087	2017-02-01 09:23:38	docs(HighchartWidget) a few comments about Highcharts initialization
d9ef4905b5230be	2017-02-01 09:23:37	fix(PA Simulations) validation errors (#PFCD-2758)
df7faa6f7243d68	2017-01-31 18:09:25	[EC] Fix for CX and PX loaddata command to work with ALL PX sized tables
31f9c8d3bfda8da	2017-01-31 15:18:14	fix(PP): fix css for normal dialog (#PFCD-2776)
6dedf7baee0e8b3	2017-01-31 10:52:13	fix(login): auto focus username (#PFCD-2715)
e8d3270bdc2071e	2017-01-31 10:25:24	fix(PP): invalidate the ds values cache after copying a table (#PFCD-2694)
da00428793be68e	2017-01-31 09:07:15	fix(missing required fields in CFS mapping) #PFCD-2779
a9ad997ee074086	2017-01-30 16:52:17	refactor(PP): remove duplicated code to int the values grids
96603f8c8499c16	2017-01-30 14:28:51	fic(PP): hide editable buttons if the user is not entitled to modify the values (#PFCD-2775)
94ac9a194db8077	2017-01-30 13:24:05	refactor(PP): reuse existing user group checking method and add an info text if the PP values grid is not editable (#PFCD-2673)
0e332f91d2bab71	2017-01-30 12:38:53	fix(PP): make the PP values grid not editable if the user is not entitle (#PFCD-2673)
eed2ee558be3f03	2017-01-27 13:58:46	fix(LPG/PGI details): move to the next product only after the user complete the interaction with it (#PFCD-2708)
bf0e556e16eac87	2017-01-26 12:57:57	feat(LPG/PGI details): preserve the selection between LPG and PGI details (#PFCD-2702)
abfc03f578e5f2e	2017-01-25 13:47:46	PFCD-2538 tiny cleanup
1e192dcf71ab689	2017-01-25 13:47:25	PFCD-2538 just set quote dto before product pikcer open, do not rebuild ui
9d1d06bc749558e	2017-01-25 13:47:17	PFCD-2538 cleanup
bbee9cdde1ae060	2017-01-25 13:46:55	PFCD-2538 quote product filter formulas ui
d7c3df2730c7ef9	2017-01-25 13:45:55	PFCD-2538 initial Quote Product Filter formulas
0d2cacc503fe2cb	2017-01-25 10:19:40	perf(PA FieldCollectionEditor): when switching collections (#PFCD-2761) Switching Datasources/Datamarts/Simulations was really slow, now it improved to somewhat slow.
644c8854957a8dc	2017-01-23 15:02:20	PFCD-2660 - minnior fixes to reduce reflow
13b106311270f52	2017-01-19 17:38:21	fix(FieldPicker) PASimulations picker missing some fields #PFCD-2739 When a field picker's source is a DM or Sim DM only underlying DS fields were listed. Now all numeric or dimension fields of the DM are there. This change also affects the query editor Customer, Product and Rebate Queries in the Configurations menu.
e02948cc6d85ce2	2017-01-19 10:51:53	feat(icons): new unload grid icons
c57a2a983530f75	2017-01-18 14:51:26	Label fix
4e8d1bfc11f0317	2017-01-17 16:07:41	refactor(login showcase): better "read more" style

f1e5bf2a3440cf5	2017-01-17 13:13:46	PFCD-2308 Unable to open/approve RR from My Pending Approvals dashlet
3d591cfa6891328	2017-01-17 13:13:30	PFCD-2732 RR email templates are missing
a587b1a916b7ba8	2017-01-16 10:13:36	refactor(customer styles): move the customer styles into a new folder outside the main
e298dbbac604161	2017-01-13 11:32:02	feat(custom css): add custom css for removing the showcase section from the login page
c00c3dc6c35dc13	2017-01-10 18:08:20	PFCD-2659 Sorting by Comment column in DCR do nothing -> remove all comments impropper filtering from filter
da092f1af32697b	2017-01-09 10:21:18	fix(pfx theme): improve text editor style (#PFCD-2703)
9ff11b970fa6ccd	2017-01-07 13:04:57	PFCD-2669 fix -> handling also errors ina logic -> small refactor
b2d82c7b4a05474	2017-01-07 12:02:53	PFCD-2669 fix - possible null pointer
90e663e1fe62180	2017-01-06 18:57:22	fix(AdvancedListGrid): clone the classic file manager selection behaviour
335e84ab826afd6	2017-01-06 13:33:34	feat(ProductDetailsEditorPortalView): refresh product details list when a item is approved or rejected (#PFCD-2679 / #PFCD-2708)
fced5653a280649	2017-01-05 22:36:09	fix(AdvancedGrid): uniform selection style in columns visibility and columns reordering grids
ff1556c1e49d44d	2017-01-05 16:35:36	PFCD-2669 It is possible to submit Contract with calculation errors and empty Required input fields -> checking for empty required fields in lineitems -> checking for line item resolution status -> checking in every lineitem editor
2a64c9653b6ddfe	2017-01-05 12:18:33	refactor(ProductDetailsEditorPortalView): make const static final
cdbd641e22e4b52	2017-01-05 10:33:11	fix(DenialReasonPickerWindow): fix i18n id for other input label
3001726e74d9aa8	2017-01-04 16:25:31	feat(AdvancedListGrid): add option to preserve the current selection between refresh (#PFCD-2679 / #PFCD-2702)
feca12e26508074	2017-01-04 13:55:25	feat(ProductDetailsEditorPortalView): scroll to the selected product on opening (#PFCD-2679 / #PFCD-2701)
db82632d7247a6d	2017-01-04 13:43:23	PFCD-2667 - autofit is now preserved, and also prefferences are applied
10d4ed6e61c0d46	2017-01-04 11:32:46	feat(AdvancedGrid): improve column reorder (#PFCD-1322)
2bf3531ecaf816d	2017-01-03 17:18:00	PFCD-2659 Sorting by Comment column in DCR do nothing -> sorting by MPL comments is allowed
15cf0a29f7bde05	2017-01-03 14:05:39	change(drag select in pde view) #PFCD-2681
6e0bbd8a6e5741c	2017-01-02 19:55:05	PFCD-2659 - explicit disable order by comments in show all mode
f2a14cac6fd6780	2017-01-02 16:56:23	refactor(combination types icons): new colors (#PFCD-2615)
64d60a86cf1ae39	2017-01-02 14:10:56	feat(login blog): new text style and position
acadfba6f1d601d	2017-01-02 12:52:26	PFCD-2685 After switch between views in DCR approve/deny buttons disappear

62c78992d4c4d83	2017-01-02 10:44:45	fix(AdvancedListGrid): fix bottom-top range selection (#PFCD-2664)
d2a96d93da644c3	2016-12-22 19:44:52	fix(page) - Opening ConfigurationPage twice causes UI to hang on "Loading data..." modal dialog (#PFCD-2641)
09d41d891a73cd4	2016-12-22 08:34:57	fix(PL config grid) #PFCD-2668 Make output grid scrollable in PL config view
58f858f7263b4ad	2016-12-21 16:57:11	fix(main nav): fix ms edge layout issue (#PFCD-2662)
b074ace7794ac68	2016-12-21 16:04:26	refactor(Login Blog): decrease the font size and move the text on top-left
b91d9137ba52837	2016-12-21 13:09:08	PFCD-2514 allow run with older server
ab540ecf992ab6b	2016-12-20 13:55:42	fix(JSConversionUtil.convertToDate): skip also function objects (#PFCD-2666)
3234fda379e568b	2016-12-20 12:09:17	fix(AdvancedListGrid): workaround to prevent the browser freezing when the grouping is enabled (#PFCD-2671)
604b46a00fe8e9c	2016-12-20 09:15:43	PFCD-2619: value from expression editor was not stored
7d8a480bdeb0679	2016-12-19 09:57:03	PFCD-2663: Data Analyzer: Cannot change type of additional measures
c0fc4e73a9bd08e	2016-12-18 15:40:04	PFCD-2667 Auto-fit, column order and sorting are reset after Show Changes Only is clicked in a DCR -> fixed by propper reordering
9ebc6a2929f9840	2016-12-17 22:05:23	PFCD-2613 - show changes are now remebered. Last choosen value is used
8dfc08c325f641d	2016-12-15 15:57:56	Fix column reorder popup title
3db48ace05cfbfb	2016-12-15 15:39:24	fix(grids): fix range selection and improve responsiveness (#PFCD-2664)
f0d55f80b0724b5	2016-12-14 16:08:11	PFCD-2625 - fix add button PFCD-2640 - fix scrolling PFCD-2613 - refactoring
da056e3f0f277af	2016-12-14 12:45:18	PFCD-2514 pdf export button available only when clouconvert enabled and xls template present
0835cea6683c89e	2016-12-13 16:05:26	Revert "PFCD-2587 Disabling event message filtering" This reverts commit a24b6664c15cea766a0002b9ad288f967520bf3a.
ee2843ef04237d8	2016-12-12 18:18:38	PFCD-2600 various PA Data Entitlements improvements - if any entitlements are defined for a DS, users with no appropriate group cannot see anything - empty filter is now a wildcard (full data acccess in DS) - allow deleting multiple rows in entitlements editor
52e7ead50b972f1	2016-12-12 16:43:20	PFCD-2250 minor label fix
197a9b2efda73b3	2016-12-08 18:59:01	PFCD-2635: Dimension filters fail on Dashboard:; in-use DS was being destroyed in DS cache mgmt
ca89af767ca4f06	2016-12-08 15:14:48	PFCD-2571 dbmigrations ui PFCD-2571 cleanup a bit PFCD-2571 two locations instaed of sql suffix for PREs PFCD-2571 rename Baseline button to Baseline V1 PFCD-2571 server migrations PFCD-2571 cleanup
59d4d7a7f1a4778	2016-12-08 14:53:55	fix(Contracts screen): Contracts: Auto refresh after go back to list page (#PFCD-2633)
5a7b262b4080f2b	2016-12-07 23:34:30	PFCD-2590 PFCD-2628 (aggregation selection from ListGridFields, i.e. chart controller and sim query editor)
78c50ec2318936f	2016-12-07 13:36:05	PFCD-2631 Show duration column in JST

2138af9428f26f5	2016-12-07 12:03:03	fix(column reorder): enable multi selection via checkboxes (#PFCD-2629)
3d4d59cd8eb7bf7	2016-12-06 15:17:16	PFCD-2556 Small array initialization fix to avoid an ugly stack trace
1e8f7623f505d69	2016-12-06 07:30:22	PFCD-2620 PFCD-2621 Fix message templates page
a88cd67e7f24caa	2016-12-05 18:59:50	PFCD-2619 Charts issue: wrong aggr. expression in additional measures Experimental, but no danger to deploy it
f1237172783afad	2016-12-05 14:11:20	PFCD-2617 Copy action not working in PA Simulations grid
35ac0c4d2ad0746	2016-12-05 11:06:49	fix(edit quote): change the item grid selection to multiple
aba364193fbb178	2016-12-02 10:28:13	PFCD-2593 Open up RRs to CFSs
f82ed55b30b275b	2016-12-01 15:44:34	merge fixe in prevoius PFCD-2590 commit
3618afcba08eb9a	2016-12-01 15:26:29	merge fix in PFCD-2590
5af74b3bf482b0d	2016-12-01 15:18:59	Missing import in previous PFCD-2590 commit
1c9906c0d18c4be	2016-12-01 14:08:29	fix(grids): preserve the selection when editing a cell (#PFCD-2603)
765be6de6b0aaa5	2016-12-01 12:28:44	refactor(DenialReasonPickerWindow): remove the wrong use of GWT.runAsync
437d9083b59013b	2016-12-01 12:26:33	fix(grids): improve double-click responsiveness
379b2c6fdc0ed96	2016-11-30 17:10:24	PFCD-2419 - LPG Detail View/Portlet View - All columns editable
261bc4e64843368	2016-11-29 14:23:48	PFCD-2452 Show also individual alerts in a way that it is not overlayed by a critical alert
3898909886c9394	2016-11-29 14:18:37	PFCD-2590: Dimensions vs measures aggregation functions in Additional Measures
ae3b143937a8f39	2016-11-29 13:35:36	PFCD-2599 Make size of configurator window formula controllable
c21910f617e921d	2016-11-29 09:35:26	PFCD-2597 PX/CX admin grid was not editable (label, user groups,...)
a24b6664c15cea7	2016-11-28 15:44:49	PFCD-2587 Disabling event message filtering
d8aab70aa93a678	2016-11-28 09:38:40	fix(DenialReasonPickerWindow): fix wrong import
67884d8fa377219	2016-11-26 19:32:05	PFCD-2205 cloudconvert system config does not work on virtual system partition
b33db78167b760c	2016-11-25 20:10:06	refactor(DenialReasonPickerWindow): improve ux

SeaBreeze 3.0 pricefx-server Changelog

- pricefx-server ChangeLog Issue Specific
- pricefx-server ChangeLog All Changes

pricefx-server ChangeLog Issue Specific

Commit	Date	Issue	Message
45ca04fa128f824	2016-06-15 10:04:41	PCFD-2019	PCFD-2019 delete associated CA on PL, MPL and PG delete
2340bda74c11658	2016-07-25 08:13:43	PFCD-1657	PFCD-1657 fix case sensitive template name 2/2
e5caf6cb16c1cc8	2016-07-25 08:13:34	PFCD-1657	PFCD-1657 fix case sensitive template name 1/2
b9066198e040a85	2016-07-25 08:13:24	PFCD-1657	PFCD-1657 Make the approval email's subject customizable
64fcf262b22cdc9	2016-05-31 14:26:33	PFCD-1934	resolved #PFCD-1934: RM: filtering on Customer(s) field in agreements landing page throws error The problem is with CustomerGroup (on RBA, CT, RR & PR) and ProductGroup (on RR & PR): the web client allows for filtering and sorting on those fields. The solution is to map those field to the CG/PG-FieldValue using the virtualField mechanism (annotation). Note; this means filtering/sorting on these fields misses objects with a filter-based CG/PG, but that's assumed to be OK as the results searching into the filter string could bring up counter-intuitive results.
c3c07ed74ad8294	2016-06-15 15:22:14	PFCD-1951	PFCD-1951 Delete last batch on Oracle & dynamic LPG
dc123003c763740	2016-06-03 10:01:38	PFCD-2037	RR.description column for PFCD-2037
7bd02616356b47b	2016-06-03 09:53:17	PFCD-2037	RR.description column for PFCD-2037
891a41da191ac70	2016-06-02 21:08:57	PFCD-2037	PFCD-2037: adding description 'lob' field to RR & allow setting it via REST API via rebaterecord.update
789abc745386b27	2016-06-13 16:36:08	PFCD-2080	PFCD-2080: Ability to filter on fields with 'Expression' : calc fields not resolved in DM virtual view
0ba01e879e568f2	2016-05-30 13:30:36	PFCD-2082	PFCD-2082: PA: query error on scatter zoom (also corrected context for filter/aggr filter, but from the web-client that would never have been a problem).
125622ded42b0ee	2016-05-30 15:41:54	PFCD-2083	PFCD-2083 contstrain generated unique key size to fit mysql innodb
832f2fe84c876ab	2016-06-01 13:06:36	PFCD-2089	PFCD-2089: PA sim-DM query broken Query generation expands the sim-DM to the underlying source-query. As sim-DMs are themselves persisted, it shold instead simply query the table.
68f5de1630ad6d5	2016-06-01 14:25:25	PFCD-2090	PFCD-2090: PA: row limit query returning error
1e70a92f9ae75a3	2016-06-02 10:53:29	PFCD-2091	PFCD-2091: a calculated field's expression is resolved - in update mode - only on the parent FC's DTO. And when updating the persisted FC form the DTO, internalExpression was not copied along. Pre-SB, the persisted FC itself was also validated (which was superfluous) so the bug didn't have an impact there.
282c2d1da141f33	2016-06-24 06:37:31	PFCD-2102	PFCD-2102 Rework open ToDo detection as it did not catch group-only scenario (and also add it to master in the first place)
cd7b7005f722db2	2016-06-09 19:31:05	PFCD-2104	PFCD-2104 - implemented backend for new help api method
647911688b3b310	2016-06-14 22:23:02	PFCD-2111	PFCD-2111: PA: incorrect 'line level' DM filter resolution (being tested..)
a03117fac3df137	2016-06-17 09:03:57	PFCD-2114	PFCD-2114 update in api.addOrUpdate does not work for instances of DynamicBusinessKeyObject
4f471ccafd4be38	2016-06-24 06:15:54	PFCD-2118	PFCD-2118 Fix permission set for "product details only" role
9566d09f2c2d572	2016-06-28 11:12:02	PFCD-2119	PFCD-2119: type in VAR_SAMP function (was VAR_SAMPLE)
f4bedd1b825bc50	2016-06-28 10:44:39	PFCD-2119	PFCD-2119: type in VAR_SAMP function (was VAR_SAMPLE)
55a64d7c80fc162	2016-06-20 16:34:29	PFCD-2119	PFCD-2119: more statistical functions in PA query
99f3c05802880ae	2016-06-22 18:45:00	PFCD-2126	PFCD-2126 Invariants configuration doesn't work with encrypted password
d11969b9523a3f7	2016-06-22 13:49:34	PFCD-2127	PFCD-2127 Invariants: Align discovery of formulas via validAfter, targetDate with rest of the application On pricing formula save action all invariants that fits into the interval of pricing formula validity are executed. Covers all possible target dates in real pricing formula executions (like in pricelists, LPGs etc.)

4191eda459fa6b6	2016-06-24 08:39:45	PFCD-2130	PFCD-2130 Fix node redirection in HZ clusters
ff6f4141e86bab2	2016-06-27 08:42:10	PFCD-2133	PFCD-2133: PA: DL for system generated DSs not created in new partitions
7fe5dca2bb75ede	2016-06-27 09:36:43	PFCD-2135	PFCD-2135 Implement submit after creation for RAT API
9380ee6055626ad	2016-06-28 10:33:20	PFCD-2142	PFCD-2142 dcr approval fails with "LTV" and "MLTV" types
02e5e5bb5d3abe0	2016-06-30 09:24:15	PFCD-2143	PFCD-2143 Add Pricelist import/export/archiving option
12ff74413d6d188	2016-06-30 08:22:29	PFCD-2143	PFCD-2143 Add Pricelist import/export/archiving option
dcf58d81a503a59	2016-07-08 11:12:53	PFCD-2146	PFCD-2146 single preferences and dashboards
4c8103b60f36d1e	2016-06-28 10:29:20	PFCD-2147	PFCD-2147 Broken SQL command line loader: second take
0714c90cbe119a7	2016-06-28 09:19:31	PFCD-2147	PFCD-2147 Broken SQL command line loader
301a40ce6087472	2016-07-13 10:25:39	PFCD-2152	PFCD-2152 Fix NPE if user has Data Manager role and requests DF data PA Data Entitlements implementation did not handle the case of Data Feed queries! Now simply ignoring entitlements for DFs, because Schema Editor and Data Manager are the only roles who have access to these anyway.
6a206d5528bf8c8	2016-07-01 10:03:01	PFCD-2156	PFCD-2156 Also check in API call if calling user is actually same
3c5343fd7e36430	2016-08-10 10:10:51	PFCD-2161	PFCD-2161: TTL for distributed calculation queues: one more TTL reset
ae56bc38f96791b	2016-08-03 10:20:47	PFCD-2161	PFCD-2161: TTL may still not be set in a STALLED queue condition
b3e25448e000503	2016-07-29 08:49:25	PFCD-2161	PFCD-2161: TTL for distributed calculation queues (merge fixes)
4d221844436b0c0	2016-07-29 08:32:40	PFCD-2161	PFCD-2161: TTL for distributed calculation queues
fded7ab36ca2fd2	2016-07-10 11:26:04	PFCD-2161	PFCD-2161: TTL for distributed calculation queues: Tighter (30min vs previous 3hr), more robust and realistic (TTL rest on each bach update, vs set at the start of the job). Note: TTL seems to get some for some queues / under some conditions, even though the code gurantees TTL setting on job start (master side).
930301f90e8f4b8	2016-07-03 16:00:41	PFCD-2161	PFCD-2161: TTL for distributed calculation queues
0ca468ce45fa37d	2016-07-03 14:49:54	PFCD-2161	PFCD-2161: TTL for distributed calculation queues
dbf711b0a9958df	2016-07-04 13:09:24	PFCD-2162	PFCD-2162 - sqlldr in PriceAPI job is failing when trying to read file with .dat extension
06a70aaee1cec56	2016-07-05 13:18:43	PFCD-2164	PFCD-2164: PA sim: reload sim-DM limits #rows to 100k
bc1421824766c68	2016-07-28 12:50:39	PFCD-2170	PFCD-2170 Fix two more issues if filling cells with multiline text 1. PoiExcelUtil was not setting appropriate cell merge range 2. DMQueryExcelClientResponse was ignoring cell merging
e31bc96e19e9183	2016-07-14 13:15:41	PFCD-2170	PFCD-2170 XLS export of PA charts: split large ChartBuilder expressions Excel cells can't hold more than 32k chartacter (or at least PoiExcelUtil enforces this). For really huge chart builder groovy expressions, we now split them into multiple rows.
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring

8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
9899c0367806bd2	2016-07-15 07:57:05	PFCD-2188	PFCD-2188 Simulation with multiple sim-only PPs cannot be converted to PL

3dad504fca73caf	2016-07-15 07:39:11	PFCD-2195	PFCD-2195 Make filters in sec/login log work
40ab986a354f8da	2016-08-09 10:11:56	PFCD-2199	PFCD-2199 LTV, MLTVs and UG bulkload PFCD-2199 fk loaddata on oracle PFCD-2199 simplify PFCD-2199 forgotten hunk
40ab986a354f8da	2016-08-09 10:11:56	PFCD-2199	PFCD-2199 LTV, MLTVs and UG bulkload PFCD-2199 fk loaddata on oracle PFCD-2199 simplify PFCD-2199 forgotten hunk
40ab986a354f8da	2016-08-09 10:11:56	PFCD-2199	PFCD-2199 LTV, MLTVs and UG bulkload PFCD-2199 fk loaddata on oracle PFCD-2199 simplify PFCD-2199 forgotten hunk
40ab986a354f8da	2016-08-09 10:11:56	PFCD-2199	PFCD-2199 LTV, MLTVs and UG bulkload PFCD-2199 fk loaddata on oracle PFCD-2199 simplify PFCD-2199 forgotten hunk
71293f44bae17d8	2016-07-18 09:30:14	PFCD-2200	PFCD-2200 Allow raw CSS in attributed result
68bcd4449ad973a	2016-07-14 13:43:40	PFCD-2203	PFCD-2203 - Update english description of password rule hints
e49b45ef3a838e5	2016-07-14 13:40:34	PFCD-2203	PFCD-2203 - Update english description of password rule hints
c3a63ee72bc8808	2016-07-18 12:49:38	PFCD-2208	PFCD-2208 Allow also retyping of name column w/o re-labeling
2b9300f52445f86	2016-07-21 12:12:28	PFCD-2220	PFCD-2220 Add token text to email
c10c593e16a1ed9	2016-07-25 07:08:39	PFCD-2222	PFCD-2222 Special role "User group override" only
7f0ed4bb4285ff3	2016-07-27 07:03:00	PFCD-2227	PFCD-2227 api.pricelist call does not take manual overrides into account
f4b339f8657f446	2016-07-28 09:53:10	PFCD-2231	PFCD-2231 Allow creation of UUID from Pricing Logic
8c6167422f9f377	2016-08-18 13:20:07	PFCD-2237	PFCD-2237 CA should be copied over in case of revision of PL
35e4755fe32c0c1	2016-07-29 10:38:13	PFCD-2238	PFCD-2238: Remove cluster wide event logging
6d7097d55a90e09	2016-07-29 10:24:50	PFCD-2238	PFCD-2238: Remove cluster wide event logging
10a1eaba9a0651f	2016-07-29 14:15:09	PFCD-2241	PFCD-2241 oracle & priceapi & date no need to write any other than full iso into price api data file
fa9962b0e2c5408	2016-08-18 13:28:22	PFCD-2247	PFCD-2247 compilation fix oops, this is embarrassing 🙂
e9a1989b9a7801f	2016-08-16 07:50:02	PFCD-2259	PFCD-2259 Add file upload permissions also to PA datamanager role
e8e1253fdf6e940	2016-08-18 15:47:12	PFCD-2267	PFCD-2267 Filter out 4-byte UTF chars from PriceAPI responses (2nd case)
b8028a765b1a82e	2016-08-18 10:19:20	PFCD-2267	PFCD-2267 Filter out 4-byte UTF chars from PriceAPI responses
0c6d55369848692	2016-09-30 06:35:09	PFCD-2312	PFCD-2312 - PriceAPI downloads taking too long - price api worker shutdown hook / Port PFCD-2312 (PriceAPI download taking long) to Sea Breeze (this commit + 2 commits cherry picked)
0c6d55369848692	2016-09-30 06:35:09	PFCD-2312	PFCD-2312 - PriceAPI downloads taking too long - price api worker shutdown hook / Port PFCD-2312 (PriceAPI download taking long) to Sea Breeze (this commit + 2 commits cherry picked)
6d3bb5b4cb00208	2016-09-30 06:26:02	PFCD-2312	PFCD-2312 PriceAPI downloads taking too long tweaks
0a8cd88d1987d34	2016-09-30 06:25:47	PFCD-2312	PFCD-2312 PriceAPI downloads taking too long - introducing per partition single threaded executor -> Customers won't block each other. Note single customer (partition) is still processing API jobs sequentially.
22bae931f738f5a	2016-10-03 18:55:25	PFCD-2318	PFCD-2318 - PX mass update/delete authorisation error - ported to Sea Breeze (task PFCD-2378)
6835cee454e0d02	2016-09-08 10:57:48	PFCD-2325	PFCD-2325 - 'Does not contain' filter not working correctly with joined tables --> fixing transformation function, to also properly handle NOT in a filter
6d032e997032948	2016-09-26 15:03:16	PFCD-2326	PFCD-2326 - Changing AttributedResultset.withBackgroundColor to null / PFCD-2359 - Port PFCD-2326 to master (to Sea Breeze)

6d032e997032948	2016-09-26 15:03:16	PFCD-2326	PFCD-2326 - Changing AttributedResultset.withBackgroundColor to null / PFCD-2359 - Port PFCD-2326 to master (to Sea Breeze)
f983d322749403c	2016-09-14 16:17:19	PFCD-2342	PFCD-2342: PA query: group level statistics not calculated correclty
a4226053fec50f5	2016-10-26 10:40:53	PFCD-2355	PFCD-2355 Fill also "last update by" as also the date is changed
6d032e997032948	2016-09-26 15:03:16	PFCD-2359	PFCD-2326 - Changing AttributedResultset.withBackgroundColor to null / PFCD-2359 - Port PFCD-2326 to master (to Sea Breeze)
93473fe46b38664	2016-09-30 08:59:39	PFCD-2367	PFCD-2367 Force numeric conversion in case queries with operators are used to get correct results. Now with floating point support
c1f770aad28f0f4	2016-09-30 07:05:14	PFCD-2367	PFCD-2367 Force numeric conversion in case queries with operators are used to get correct results
22bae931f738f5a	2016-10-03 18:55:25	PFCD-2378	PFCD-2318 - PX mass update/delete authorisation error - ported to Sea Breeze (task PFCD-2378)
aaa30fd6b4e34a8	2016-10-04 10:49:00	PFCD-2380	PFCD-2380 disable scientific notation interpretation no mysql
e87bf5380864b22	2016-10-26 10:39:40	PFCD-2389	PFCD-2389 NPE check, boxed type
b438e62720e1315	2016-10-26 10:39:18	PFCD-2389	PFCD-2389 exclude submitter from mail sending to approvers
bf255fa428e3106	2016-10-10 11:15:39	PFCD-2394	PFCD-2394 - Allow (whitelist) PCOMP data modification Class ProductCompetition now implements APIModifiable interface.
d1c13851a4de2ff	2016-10-17 09:49:15	PFCD-2405	PFCD-2405 DMDataSource import from file fix
32fd2e15ec5666c	2016-11-01 07:42:02	PFCD-2422	PFCD-2422 npe on data export of noncalculated entities
bacaf185a9777b3	2016-10-21 14:35:58	PFCD-2451	PFCD-2451 JSON serialize sql timestamps including time part
0733a402662f57c	2016-10-21 13:40:14	PFCD-2451	PFCD-2451 (REVERT) JSON serialize sql timestamps including time part
0469cd4b1913221	2016-10-20 09:04:13	PFCD-2451	PFCD-2451 JSON serialize sql timestamps including time part
eac5775aa6dc812	2016-10-28 05:32:17	PFCD-2478	PFCD-2478 Force recalculation on preview to make sure the workflow formula has full set of data

pricefx-server ChangeLog All Changes

Commit	Date	Message
4b8dee6ca48975a	2016-11-15 08:43:29	Minor compilation fix
59efc8bc23d040b	2016-11-15 08:34:23	Rework PGI event data generation Add jetty compression support (default: off)
32fd2e15ec5666c	2016-11-01 07:42:02	PFCD-2422 npe on data export of noncalculated entities
eac5775aa6dc812	2016-10-28 05:32:17	PFCD-2478 Force recalculation on preview to make sure the workflow formula has full set of data
2984e8861d6eda8	2016-10-27 11:13:10	Export completeCalculationResults also for ITEM_UPDATE_PGI event
a4226053fec50f5	2016-10-26 10:40:53	PFCD-2355 Fill also "last update by" as also the date is changed
e87bf5380864b22	2016-10-26 10:39:40	PFCD-2389 NPE check, boxed type

b438e62720e1315	2016-10-26 10:39:18	PFCD-2389 exclude submitter from mail sending to approvers
5f72c60bd2df09e	2016-10-24 12:21:59	Export completeCalculationResults instead of activeCalculationResults as part of the events.
fe72bf3dfa1a692	2016-10-24 09:09:18	Also export activeCalculationResults as part of the events. Lower the retainment period for sucessful events to 5 days to compensate
2cf902505ee37a9	2016-10-21 14:50:51	Fix obsolete test
bacaf185a9777b3	2016-10-21 14:35:58	PFCD-2451 JSON serialize sql timestamps including time part
0733a402662f57c	2016-10-21 13:40:14	PFCD-2451 (REVERT) JSON serialize sql timestamps including time part
0469cd4b1913221	2016-10-20 09:04:13	PFCD-2451 JSON serialize sql timestamps including time part
d1c13851a4de2ff	2016-10-17 09:49:15	PFCD-2405 DMDataSource import from file fix
58bad6ac18ae65f	2016-10-13 07:18:13	Add some more SAML logging
bf255fa428e3106	2016-10-10 11:15:39	PFCD-2394 - Allow (whitelist) PCOMP data modification Class ProductCompetition now implements APIModifiable interface.
e3166d8ea6315d2	2016-10-07 07:24:34	Set debug on redis stuff lower for now by default
3ebb14ac1134ffd	2016-10-05 07:38:25	Better logging for SQL*Loader configuration issues
aaa30fd6b4e34a8	2016-10-04 10:49:00	PFCD-2380 disable scientific notation interpretation no mysql
22bae931f738f5a	2016-10-03 18:55:25	PFCD-2318 - PX mass update/delete authorisation error - ported to Sea Breeze (task PFCD-2378)
b0ef557ec8372bb	2016-10-03 12:13:01	Force new session in saml consume (to prevent session reuse)
93473fe46b38664	2016-09-30 08:59:39	PFCD-2367 Force numeric conversion in case queries with operators are used to get correct results. Now with floating point support
c1f770aad28f0f4	2016-09-30 07:05:14	PFCD-2367 Force numeric conversion in case queries with operators are used to get correct results
0c6d55369848692	2016-09-30 06:35:09	PFCD-2312 - PriceAPI downloads taking too long - price api worker shutdown hook / Port PFCD-2312 (PriceAPI download taking long) to Sea Breeze (this commit + 2 commits cherry picked)
6d3bb5b4cb00208	2016-09-30 06:26:02	PFCD-2312 PriceAPI downloads taking too long tweaks
0a8cd88d1987d34	2016-09-30 06:25:47	PFCD-2312 PriceAPI downloads taking too long - introducing per partition single threaded executor -> Customers won't block each other. Note single customer (partition) is still processing API jobs sequentially.
6d032e997032948	2016-09-26 15:03:16	PFCD-2326 - Changing AttributedResultset.withBackgroundColor to null / PFCD-2359 - Port PFCD-2326 to master (to Sea Breeze)
f983d322749403c	2016-09-14 16:17:19	PFCD-2342: PA query: group level statistics not calculated correclty
6d89eb2019335cf	2016-09-11 21:52:36	PA sim fix: when using regular (PB) formula, outputs not mapped (and hence populated) correclty
5fdf69c2650da09	2016-09-11 20:00:01	PA sim fix: need to also be able to evaluate regular (PB) formulas

8dc303747d34f0e	2016-09-11 19:55:33	PQ query --< fx fix: was not leveraging all funcitonal dependencies when determining FC key fields
e4fcc26da794581	2016-09-11 19:54:53	PA sim fix: need to also be able to evaluate regular (PB) formulas
4d812a4efe815c5	2016-09-11 19:54:02	PA sim fix: was unable to select regular (PB) formulas
6835cee454e0d02	2016-09-08 10:57:48	PFCD-2325 - 'Does not contain' filter not working correctly with joined tables --> fixing transformation function, to also properly handle NOT in a filter
b572b67be90908e	2016-09-07 16:50:57	PA actions: always closing the hibernate session (possily a facotr in Bosch connections issue)
e1c187e45d63d57	2016-09-07 15:44:15	bf: IndexFilter not applied ('Index' function from chart UI)
5f1aad900da4d13	2016-09-02 11:34:39	Changed from Response to FetchResponse
c72d64051bd038d	2016-08-23 05:49:55	Swagger API: added Datamart.updatefc and getDataLoads
050f5a1ef5dd8ff	2016-08-19 16:21:26	Fully initialize CLI object before returning
e8e1253fdf6e940	2016-08-18 15:47:12	PFCD-2267 Filter out 4-byte UTF chars from PriceAPI responses (2nd case)
fa9962b0e2c5408	2016-08-18 13:28:22	PFCD-2247 compilation fix oops, this is embarrassing 🙂
8c6167422f9f377	2016-08-18 13:20:07	PFCD-2237 CA should be copied over in case of revision of PL
b8028a765b1a82e	2016-08-18 10:19:20	PFCD-2267 Filter out 4-byte UTF chars from PriceAPI responses
e6f5e64d7098892	2016-08-17 06:56:53	Fix incorrect type of Calculation Summary message for simulations
e9a1989b9a7801f	2016-08-16 07:50:02	PFCD-2259 Add file upload permissions also to PA datamanager role
441894c3d811a66	2016-08-11 07:10:40	Fix sim-only lookup tables not displayed during conversion to pricelist
415226016fe7215	2016-08-10 19:40:22	NPE on stand-alone DM update
3c5343fd7e36430	2016-08-10 10:10:51	PFCD-2161: TTL for distributed calculation queues: one more TTL reset
708a9e026db9f96	2016-08-10 10:09:56	Adding BoneCP connectionTimeoutInMs param (new connections) with default of 15min (to be lowered at a later time).
40ab986a354f8da	2016-08-09 10:11:56	PFCD-2199 LTV, MLTVs and UG bulkload PFCD-2199 fk loaddata on oracle PFCD-2199 simplify PFCD-2199 forgotten hunk
cdb62f12f95d72f	2016-08-08 14:29:29	Swagger API - datamart.getfcs transaction added
9d528a663594b0b	2016-08-05 15:07:19	[EC] Upgrade to Excel-DNA version 0.33.9
7b50800ea023baf	2016-08-05 12:34:28	Fixing shutdown sequence
ae56bc38f96791b	2016-08-03 10:20:47	PFCD-2161: TTL may still not be set in a STALLED queue condition

10a1eaba9a0651f	2016-07-29 14:15:09	PFCD-2241 oracle & priceapi & date no need to write any other than full iso into price api data file
35e4755fe32c0c1	2016-07-29 10:38:13	PFCD-2238: Remove cluster wide event logging
6d7097d55a90e09	2016-07-29 10:24:50	PFCD-2238: Remove cluster wide event logging
b3e25448e000503	2016-07-29 08:49:25	PFCD-2161: TTL for distributed calculation queues (merge fixes)
4d221844436b0c0	2016-07-29 08:32:40	PFCD-2161: TTL for distributed calculation queues
bc1421824766c68	2016-07-28 12:50:39	PFCD-2170 Fix two more issues if filling cells with multiline text 1. PoiExcelUtil was not setting appropriate cell merge range 2. DMQueryExcelClientResponse was ignoring cell merging
67822b7ba4fd1c5	2016-07-28 09:56:13	Fix refactoring regression
f4b339f8657f446	2016-07-28 09:53:10	PFCD-2231 Allow creation of UUID from Pricing Logic
45dc2c0d1e7f96b	2016-07-27 10:41:12	Take redis out of normal logins (use local user ban maps by default)
bdfdf6075184e2a	2016-07-27 08:15:26	Fix log rotation for per-partition logs
7f0ed4bb4285ff3	2016-07-27 07:03:00	PFCD-2227 api.pricelist call does not take manual overrides into account
445a49aac74b180	2016-07-26 12:06:10	Swagger API: support for business key (new in SB) in ProductExtensionDefinition
8d262fcb5c53cca	2016-07-25 12:01:17	PFCD-2173 fix sqlldr and clobs on oracle PFCD-2173 typos PFCD-2173 accept only latest records when overwrite=true PFCD-2173 typo PFCD-2173 direct path and tempTargetId instead of rewriting id in target table PFCD-2173 forgotten changeset PFCD-2173 forgotten ids on input table when doing direct load PFCD-2173 check doc strings and narrow filter when handling clobs PFCD-2173 sonar wishes PFCD-2173 10M buffer as default for lob field when doing bulkload PFCD-2173 remove unused code PFCD-2173 use old (possibly faster) path for case when there is no clob field + tiny refactoring PFCD-2173 fix fuzzy refactoring
308f8fdcc527baa	2016-07-25 09:22:51	Introduce request UUID in logging to easier correlate requests accross log files
2340bda74c11658	2016-07-25 08:13:43	PFCD-1657 fix case sensitive template name 2/2
e5caf6cb16c1cc8	2016-07-25 08:13:34	PFCD-1657 fix case sensitive template name 1/2
b9066198e040a85	2016-07-25 08:13:24	PFCD-1657 Make the approval email's subject customizable
c10c593e16a1ed9	2016-07-25 07:08:39	PFCD-2222 Special role "User group override" only
d6be820e8c10378	2016-07-25 06:44:26	Allow to force hosted mode
4ab92300d138a37	2016-07-25 06:35:55	Do not check for ban for noSession=true users (to reduce cluster/redis traffic). Can be overridden
856e66563353fa4	2016-07-22 19:14:43	oracle patch 19632480 for 12.1.0.2.0 driver should fix leaks when asking for metadata
2b9300f52445f86	2016-07-21 12:12:28	PFCD-2220 Add token text to email
c3a63ee72bc8808	2016-07-18 12:49:38	PFCD-2208 Allow also retyping of name column w/o re-labeling

71293f44bae17d8	2016-07-18 09:30:14	PFCD-2200 Allow raw CSS in attributed result
9899c0367806bd2	2016-07-15 07:57:05	PFCD-2188 Simulation with multiple sim-only PPs cannot be converted to PL
3dad504fca73caf	2016-07-15 07:39:11	PFCD-2195 Make filters in sec/login log work
68bcd4449ad973a	2016-07-14 13:43:40	PFCD-2203 - Update english description of password rule hints
e49b45ef3a838e5	2016-07-14 13:40:34	PFCD-2203 - Update english description of password rule hints
e31bc96e19e9183	2016-07-14 13:15:41	PFCD-2170 XLS export of PA charts: split large ChartBuilder expressions Excel cells can't hold more than 32k chartacter (or at least PoiExcelUtil enforces this). For really huge chart builder groovy expressions, we now split them into multiple rows.
301a40ce6087472	2016-07-13 10:25:39	PFCD-2152 Fix NPE if user has Data Manager role and requests DF data PA Data Entitlements implementation did not handle the case of Data Feed queries! Now simply ignoring entitlements for DFs, because Schema Editor and Data Manager are the only roles who have access to these anyway.
fded7ab36ca2fd2	2016-07-10 11:26:04	PFCD-2161: TTL for distributed calculation queues: Tighter (30min vs previous 3hr), more robust and realistic (TTL rest on each bach update, vs set at the start of the job). Note: TTL seems to get some for some queues / under some conditions, even though the code gurantees TTL setting on job start (master side).
dcf58d81a503a59	2016-07-08 11:12:53	PFCD-2146 single preferences and dashboards
7b0378b153be41f	2016-07-07 13:17:30	C4BEU-114 - adjusting file to be usable in sqlplus
6a898c9dcebccb8	2016-07-07 07:42:09	fix priceapi
dba67afb20f728e	2016-07-06 08:40:55	Adding utility command to detect stale data
e49e6b3f5b44046	2016-07-06 08:09:37	ClusterNode config-auto-update fix
06a70aaee1cec56	2016-07-05 13:18:43	PFCD-2164: PA sim: reload sim-DM limits #rows to 100k
fad59e233cb6ba8	2016-07-05 08:55:02	NPE fix
dbf711b0a9958df	2016-07-04 13:09:24	PFCD-2162 - sqlldr in PriceAPI job is failing when trying to read file with .dat extension
bcdf4e1de21812c	2016-07-04 06:04:41	Also allow _ and - in metric names
930301f90e8f4b8	2016-07-03 16:00:41	PFCD-2161: TTL for distributed calculation queues
0ca468ce45fa37d	2016-07-03 14:49:54	PFCD-2161: TTL for distributed calculation queues
6a206d5528bf8c8	2016-07-01 10:03:01	PFCD-2156 Also check in API call if calling user is actually same
5476f1fa1e7a6b8	2016-06-30 13:23:00	In case no password expiry interval is set: wipe expiry date on change
4e592963a64e590	2016-06-30 10:31:36	Make archive task also work with super large sets
02e5e5bb5d3abe0	2016-06-30 09:24:15	PFCD-2143 Add Pricelist import/export/archiving option

12ff74413d6d188	2016-06-30 08:22:29	PFCD-2143 Add Pricelist import/export/archiving option
9566d09f2c2d572	2016-06-28 11:12:02	PFCD-2119: type in VAR_SAMP function (was VAR_SAMPLE)
f4bedd1b825bc50	2016-06-28 10:44:39	PFCD-2119: type in VAR_SAMP function (was VAR_SAMPLE)
9380ee6055626ad	2016-06-28 10:33:20	PFCD-2142 dcr approval fails with "LTV" and "MLTV" types
4c8103b60f36d1e	2016-06-28 10:29:20	PFCD-2147 Broken SQL command line loader: second take
0714c90cbe119a7	2016-06-28 09:19:31	PFCD-2147 Broken SQL command line loader
62c28cc227fe085	2016-06-28 08:46:07	additional logging for sqlldr problems
fe1009c4884c504	2016-06-27 10:25:26	better logging for SQLLoader - troubleshooting Bosch
7fe5dca2bb75ede	2016-06-27 09:36:43	PFCD-2135 Implement submit after creation for RAT API
ff6f4141e86bab2	2016-06-27 08:42:10	PFCD-2133: PA: DL for system generated DSs not created in new partitions
d2ee962d9a14c6d	2016-06-24 13:32:51	Missing #threads (per partition) to execute PA queries, data loads...
29bd9e71c07f189	2016-06-24 13:06:33	enable sqlldr log level switch
4191eda459fa6b6	2016-06-24 08:39:45	PFCD-2130 Fix node redirection in HZ clusters
282c2d1da141f33	2016-06-24 06:37:31	PFCD-2102 Rework open ToDo detection as it did not catch group-only scenario (and also add it to master in the first place)
4f471ccafd4be38	2016-06-24 06:15:54	PFCD-2118 Fix permission set for "product details only" role
99f3c05802880ae	2016-06-22 18:45:00	PFCD-2126 Invariants configuration doesn't work with encrypted password
d11969b9523a3f7	2016-06-22 13:49:34	PFCD-2127 Invariants: Align discovery of formulas via validAfter, targetDate with rest of the application On pricing formula save action all invariants that fits into the interval of pricing formula validity are executed. Covers all possible target dates in real pricing formula executions (like in pricelists, LPGs etc.)
6e726eb8ffaaa96	2016-06-22 10:59:01	Invariants: Align discovery of formulas via validAfter, targetDate with rest of the application On pricing formula save action all invariants that fits into the interval of pricing formula validity are executed. Covers all possible target dates in real pricing formula executions (like in pricelists, LPGs etc.)
9354e6e2a77e2f7	2016-06-21 20:33:57	Typo in postgres stddev_samp function (was stddev_sample) + Sonar
380f2b4c79c5e9c	2016-06-21 14:39:33	Typo in postgres stddev_samp function (was stddev_sample)
0dad592dee7d1f5	2016-06-21 07:49:12	DWTable for Oracle migr --> 3.0
55a64d7c80fc162	2016-06-20 16:34:29	PFCD-2119: more statistical functions in PA query
d8aa0ac149e2c78	2016-06-20 14:23:52	Minor logging fix
0b4a823ed7cf648	2016-06-20 10:26:03	hf: DM data fetch: getting NPEs after Bosch (Q) upgrade to SB

12fabd959d3a178	2016-06-17 21:49:57	Fix: was not listening for generic distributed actions (like ReloadPartitionConfig), and hence these were not executed.
9bbdd946b801110	2016-06-17 14:47:19	Swagger API: added valueFields in the FetchRequest (upon request from Tomas Brejla) for distinct queries.
7636899a4905270	2016-06-17 13:46:41	Also allow absolute relayState URLs
a03117fac3df137	2016-06-17 09:03:57	PFCD-2114 update in api.addOrUpdate does not work for instances of DynamicBusinessKeyObject
9fd59eb52623e0a	2016-06-16 12:48:52	Commit every column wipe individually to avoid /tmp file overshot
5b97de3af3fcc72	2016-06-16 07:11:37	Make SAML response check slightly less strict to allow for proxying.
d74900517177a08	2016-06-16 06:56:10	Fix build - take2
c74da5ca5e5c939	2016-06-16 06:54:12	Fix build
ae5e07d67664169	2016-06-16 06:33:13	Make SAML response check slightly less strict to allow for proxying. Add logging
a02ef3f950b78a9	2016-06-15 19:16:27	PA query fixes: start/end row ignored, query alias for clarity in source (sub-) queries...
c3c07ed74ad8294	2016-06-15 15:22:14	PFCD-1951 Delete last batch on Oracle & dynamic LPG
aa7eb4777eed11a	2016-06-15 13:00:06	Fix missing partition filter for stuff happening pre-PartitionFilter
45ca04fa128f824	2016-06-15 10:04:41	PCFD-2019 delete associated CA on PL, MPL and PG delete
647911688b3b310	2016-06-14 22:23:02	PFCD-2111: PA: incorrect 'line level' DM filter resolution (being tested..)
789abc745386b27	2016-06-13 16:36:08	PFCD-2080: Ability to filter on fields with 'Expression' : calc fields not resolved in DM virtual view
69440a237e5bb83	2016-06-10 09:13:17	Make autoApprove AND approvalTrigger play together. Hence process approvalTriggers first and only then auto approve
f2cb30329c9aa70	2016-06-09 23:27:22	PA query: formula projections were not working with defautl values (for ex. used by the WF chart)
cd7b7005f722db2	2016-06-09 19:31:05	PFCD-2104 - implemented backend for new help api method
ff72c17bc3435f1	2016-06-08 09:03:19	Adding issing Oracle migration statements
54624d359823cc7	2016-06-08 07:57:12	Sonar....
873fdf487082713	2016-06-07 15:24:22	PA dataload jobs: avoiding ObjectLockedExceptions (on target FC) by relying on JST's concurrenyKey (tragteFC.sourceName vs. DL.typedId)
897d3b895bb1312	2016-06-07 13:30:07	missing i18n rebate label
f87058cc76d4b4a	2016-06-06 22:47:14	Quick fix: all fiels ina sim-DM are to be owned by that DM, so noadditional tables should be joind into it at query time. Will need to more comprehensively review the regular vs sim DM behaviour.
6f7f0d1c502d1b2	2016-06-06 19:26:18	Regression: FCs were no longer validated in the DMAction context (spearate thread to the caller thread). It's not always required but best to align back to 2.9 when there were no issues in this area.

e4161e3c9effe2f	2016-06-03 13:24:43	Regression in 3.0: revalidating FC after update, as fields were not set, or set correclty (internalExpr, possily others).
94397ff270510dc	2016-06-03 12:20:22	no message
dc123003c763740	2016-06-03 10:01:38	RR.description column for PFCD-2037
7bd02616356b47b	2016-06-03 09:53:17	RR.description column for PFCD-2037
891a41da191ac70	2016-06-02 21:08:57	PFCD-2037: adding description 'lob' field to RR & allow setting it via REST API via rebaterecord.update
24c7dc8b4c9bf89	2016-06-02 18:28:30	Sonar warnings...
bba5082e429c7d4	2016-06-02 12:22:53	Resolving Sonar warnings....
2da2cc115883803	2016-06-02 10:56:12	Some Sonar fixes
1e70a92f9ae75a3	2016-06-02 10:53:29	PFCD-2091: a calculated field's expression is resolved - in update mode - only on the parent FC's DTO. And when updating the persisted FC form the DTO, internalExpression was not copied along. Pre-SB, the persisted FC itself was also validated (which was superfluous) so the bug didn't have an impact there.
ceb7f8510a0eed4	2016-06-01 17:03:40	Redis Jett session TTL fix
fd66cc902e595e8	2016-06-01 16:57:52	Redis Jetty sessions TTL fix
54b5028d5365f29	2016-06-01 16:43:34	Redis Jetty sessions TTL fix
ae6fb23a3487a52	2016-06-01 14:29:04	FormulaSanboxL CalculableLineItem must not be white-listed as it's a PO.
68f5de1630ad6d5	2016-06-01 14:25:25	PFCD-2090: PA: row limit query returning error
832f2fe84c876ab	2016-06-01 13:06:36	PFCD-2089: PA sim-DM query broken Query generation expands the sim-DM to the underlying source-query. As sim-DMs are themselves persisted, it shold instead simply query the table.
64fcf262b22cdc9	2016-05-31 14:26:33	resolved #PFCD-1934: RM: filtering on Customer(s) field in agreements landing page throws error The problem is with CustomerGroup (on RBA, CT, RR & PR) and ProductGroup (on RR & PR): the web client allows for filtering and sorting on those fields. The solution is to map those field to the CG/PG-FieldValue using the virtualField mechanism (annotation). Note; this means filtering/sorting on these fields misses objects with a filter-based CG/PG, but that's assumed to be OK as the results searching into the filter string could bring up counter-intuitive results.
7c5d2d06e737b9c	2016-05-31 08:54:18	PA DL: currenItem.incLoadDate in formula ctx now returns DateTime (vs timeless Date prevoiusly). We need the time portion to be able to implement in-day incremental behavoiur.
125622ded42b0ee	2016-05-30 15:41:54	PFCD-2083 contstrain generated unique key size to fit mysql innodb
0ba01e879e568f2	2016-05-30 13:30:36	PFCD-2082: PA: query error on scatter zoom (also corrected context for filter/aggr filter, but from the web-client that would never have been a problem).

SeaBreeze 3.0 pricefx-web Changelog

- pricefx-web ChangeLog Issue Specific
- pricefx-web ChangeLog All Changes

pricefx-web ChangeLog Issue Specific

Commit	Date	Issue	Message

05e1b451b37bf27	2016-07-01 13:15:55	PFCD-1301	PFCD-1301 Custom number format in PA data grids not applied as expected Was done according to column number, so applying wrong format if user moved columns around.
292f8b04591a37b	2016-07-12 13:04:30	PFCD-1413	PFCD-1413 Don't allow "limit to" filter for Boxplots For charts that support embedded Boxplots, the "limit to" and "boxplot" options are now mutually exclusive.
d5558f188fb0abb	2016-06-29 14:34:58	PFCD-1526	PFCD-1526 Open multiple PX/CX tabs
4d21a69047a8e35	2016-07-25 08:15:06	PFCD-1657	PFCD-1657 Make the approval email's subject customizable
3d75055a58bd64f	2016-05-30 12:31:00	PFCD-1807	PFCD-2080: Ability to filter on fields with 'Expression' PFCD-1807: Datasource UI - editing manually record results in duplicate row
6346cdc973d7f15	2016-07-19 10:59:01	PFCD-2001	PFCD-2001 don't fire showData in prefs after callback on console init
7108ebbc9b650d0	2016-06-02 21:11:15	PFCD-2037	PFCD-2037: fixing 'Rebate Text' widget, now showing and allowing to change RR.description
71ba21bf8b8ae64	2016-06-01 14:45:32	PFCD-2048	PFCD-2048 AdvancedGrid delete button visible only if a row is selected This is usually handled in classes using or extending Advanced(List)Grid but it was still missing in so many places that it is now handled at the AdvancedGrid level.
fd5c1d8cd52a7a0	2016-06-15 07:46:08	PFCD-2054	PFCD-2054 Hide action buttons of DM DataEditor until a source is set
3ccf555f44a6672	2016-05-31 15:35:08	PFCD-2055	PFCD-2055: Copy of scheduled job leads to an error
6f9124dcb6d354a	2016-06-10 07:31:30	PFCD-2063	PFCD-2063 Show/hide LPG approve/deny buttons in a more consistent way The same logic is now used to decide whether approve/deny buttons should appear on the row of an LPG item, and at the bottom of the grid if that same row is selected.
05f0d4a9c261998	2016-06-02 12:32:13	PFCD-2070	PFCD-2070: PA: do not show 'PerUnit' & 'Percentage' aggregation functions in Chart controller if no basis field set in the DM
dcacedba713390e	2016-05-30 13:49:48	PFCD-2077	PFCD-2077 Configurator for Logic doesn't work correctly -> editorExitHandler is ommitged for matrix widget.
32f3ac6b771a7bb	2016-06-13 16:08:03	PFCD-2080	PFCD-2080: Ability to filter on fields with 'Expression' : calc fields do not show up in FilterBuilder
3d75055a58bd64f	2016-05-30 12:31:00	PFCD-2080	PFCD-2080: Ability to filter on fields with 'Expression' PFCD-1807: Datasource UI - editing manually record results in duplicate row
497478f229046a1	2016-07-07 07:10:25	PFCD-2094	PFCD-2094 Make list grid item adjust width
b6553f23f99f914	2016-06-07 14:03:58	PFCD-2100	PFCD-2100: PA: init/row/summary group options missing in Query Calculation Logic (formula editor)
03baecfa3b5e84b	2016-07-08 11:06:57	PFCD-2105	PFCD-2105 fix ResultMatrix filtering and make sure refresh is ok in dashboards
093233c61084e01	2016-07-28 12:15:00	PFCD-2108	PFCD-2108 Paramater details are not imported when using import/export groovy
d176c10c0eaaf92	2016-06-23 11:40:52	PFCD-2109	PFCD-2109 Simple and advanced filters sometimes active at the same Sometimes the AdvancedFilterBuilder and the basic GWT filter row at the top of the grid could be displayed simultaneously. Was due to the fact that after filterBuilder.setVisible(true), filterBuilder.isVisible() could still return false if the grid had not been drawn yet.
9eb07c9a63eda67	2016-06-16 08:40:40	PFCD-2110	PFCD-2110 do not ever hide first multitier input param
e46342ad52d6e8e	2016-06-16 14:40:44	PFCD-2112	PFCD-2112 Remove useless copy action from Rebate Records grid
ad660bb576b5140	2016-06-17 14:06:30	PFCD-2118	PFCD-2118 Show edit details button also for product details -only editors
ca79de1e58f8720	2016-07-08 11:07:18	PFCD-2123	PFCD-2123 minimized saved portlets are loaded not minimized on dashboards
57c6bfc10fe5930	2016-06-23 07:03:45	PFCD-2124	PFCD-2124 MultiTierEntry hint has huge width
7d6e93fe6ed3ce6	2016-06-29 09:12:14	PFCD-2139	PFCD-2139 do make start page grids selectable
df8d356e0e045d9	2016-06-30 08:23:25	PFCD-2143	PFCD-2143 Add Pricelist import/export/archiving option

d53327d7f6185c8	2016-07-08 11:08:01	PFCD-2146	PFCD-2146 update prefs after save
71ce4e796eeff63	2016-07-08 11:07:45	PFCD-2146	PFCD-2146 invalidate prefs cache on prefs save in dashboards
6716f65e609052f	2016-07-08 11:07:33	PFCD-2146	PFCD-2146 delete pref button disabled after prefrences successfuly saved
502a2de7b31d323	2016-07-08 11:06:41	PFCD-2146	PFCD-2146 single preferences and dashboards
a272f576ed2bdb2	2016-07-18 18:11:40	PFCD-2151	PFCD-2151 Update filter controller even when filter set to null Also added one more FilterController.setFilter() method to keep the controller's link title consistent between its initial title, its title after reset, and its title when set to null.
e1801e3761165f9	2016-07-18 18:05:48	PFCD-2151	PFCD-2151 Make sure to save a Data Load's filter when it's set to null
81fd3fec6f5564a	2016-07-13 07:44:19	PFCD-2152	PFCD-2152 Fix a regression with Data Feed and Datamart data tables For users without the SchemaEditor role, the data tables were not displayed anymore.
d1550ceea4217f8	2016-07-13 07:39:49	PFCD-2152	PFCD-2152 show only relevant action buttons in DF, DS and DM lists Delete button was still showing up sometimes, if a user without SchemaEditor role selected a Data Source
e6c7885ff4e75ac	2016-07-11 12:07:56	PFCD-2152	PFCD-2152 show only relevant action buttons in DF, DS and DM lists Show/hide the "add", "delete", and "import/export" action buttons in the left hand side lists of DFs, DSes, DMs, and SimDMs according to user's role and to current selection.
1ac1eceddb8dc51	2016-07-06 09:23:08	PFCD-2152	PFCD-2152 PA: Hide deploy button + lock field editor if user not admin
4bebddc5fada556	2016-07-01 13:15:55	PFCD-2154	PFCD-2154 PA Data Table group labels sometimes disappearing
749573fa4c9109a	2016-07-19 11:32:05	PFCD-2172	PFCD-2172 Also apply in-flight/pending input value changes when hitting "price quote" (i.e. complete edit before sending quote object) Backing out approach again
02e1e2108668786	2016-07-19 11:11:15	PFCD-2172	PFCD-2172 Also apply in-flight/pending input value changes when hitting "price quote" (i.e. complete edit before sending quote object) Missing import
b58c512005754c7	2016-07-19 11:07:31	PFCD-2172	PFCD-2172 Also apply in-flight/pending input value changes when hitting "price quote" (i.e. complete edit before sending quote object)
63c2070be1983ad	2016-07-19 07:29:42	PFCD-2172	PFCD-2172 Also apply in-flight/pending input value changes when hitting "price quote" (i.e. complete edit before sending quote object)
6ff38ea7004c40e	2016-08-10 11:54:50	PFCD-2174	PFCD-2174 - fixes ambigous method call
138959399e3c36b	2016-07-14 11:13:59	PFCD-2191	PFCD-2191 - Gauge cut off in QC UI
a54d460766f8b8f	2016-07-15 07:44:14	PFCD-2194	PFCD-2194 Add download button to security/login trail logs
d56d174066b6e28	2016-07-14 12:18:31	PFCD-2197	PFCD-2197 fix ResultMatrix in single dashboard layout BEWARE! this single dashboard layout steals state from its inner portlets!
a209e88681bcab8	2016-07-19 16:31:52	PFCD-2215	PFCD-2215 - implemented cache eviction and full row render
0d6287035a98e0f	2016-07-25 07:14:08	PFCD-2222	PFCD-2222 Special role "User group override" only
87405650ec2af8d	2016-07-27 13:51:03	PFCD-2229	PFCD-2229 Dashboard controller doesn't open deep links (typedId)
7b7f217e2e75f32	2016-07-29 07:08:36	PFCD-2236	PFCD-2236 - Show/hide column / apply prefferences / reorder should be faster and show modal mask when applied -> mask is shown when changing layout, only for LPG now -> render grid is not used anymore -> creted custom vibility menu - used for custom event that properly cleares caches
463cfff4bcbf782	2016-08-02 11:35:39	PFCD-2239	PFCD-2239 - initial mask is skipped - it is distracting
a95686f1d9e0eb6	2016-08-02 07:20:38	PFCD-2239	PFCD-2239 - sonar fixes
61affbcf6360952	2016-08-02 07:10:57	PFCD-2239	PFCD-2239 - fixing tabs and added modal mask also for autofit
1deca8492fc05b0	2016-08-08 10:58:44	PFCD-2240	PFCD-2240: LPG: disable the row(s) and hide the relative approve/deny buttons until the approve or deny actions are in progress

0e3739bad9b9c7b	2016-08-03 08:25:20	PFCD-2244	PFCD-2244 Add "part_no" option into PriceAPI
659f549cba5f5ad	2016-08-10 11:54:50	PFCD-2247	PFCD-2249 Calendar buttons disappeared after PFCD-2247 The SGWT update to fix PFCD-2247 broke our EditableMiniDateRangeItem, a customized version of the default form item that shows up in the date cells of the filter row at the top of a grid.
659f549cba5f5ad	2016-08-10 11:54:50	PFCD-2247	PFCD-2249 Calendar buttons disappeared after PFCD-2247 The SGWT update to fix PFCD-2247 broke our EditableMiniDateRangeItem, a customized version of the default form item that shows up in the date cells of the filter row at the top of a grid.
71272b620b016c2	2016-08-10 11:54:50	PFCD-2247	PFCD-2247 Bump SmartGWT to build 6.0p-2016-08-04 Known regression linked to this commit: PFCD-2249
65445f6bd32c50e	2016-08-29 18:54:23	PFCD-2248	PFCD-2248 upgrade sgwt to 6.0p-2016-08-27
e829c3d06cb21d9	2016-08-26 11:29:01	PFCD-2248	PFCD-2248 - implementing spinner wheel and adding try catch block to ensure mask is turned down when task ended correctly or not
90deb6ff03ee2b0	2016-08-10 11:54:50	PFCD-2249	PFCD-2249 - set the icon size to the original one
659f549cba5f5ad	2016-08-10 11:54:50	PFCD-2249	PFCD-2249 Calendar buttons disappeared after PFCD-2247 The SGWT update to fix PFCD-2247 broke our EditableMiniDateRangeItem, a customized version of the default form item that shows up in the date cells of the filter row at the top of a grid.
71272b620b016c2	2016-08-10 11:54:50	PFCD-2249	PFCD-2247 Bump SmartGWT to build 6.0p-2016-08-04 Known regression linked to this commit: PFCD-2249
457068b05779548	2016-08-11 11:45:17	PFCD-2251	PFCD-2251 - upgrading to sgwt with fix
5013f68e489025f	2016-08-10 11:54:50	PFCD-2251	PFCD-2251 4 more validators have same issue for "no dupe" implementation
18aa051a1f33ff9	2016-08-10 11:54:50	PFCD-2251	PFCD-2251 - changed validator to not use equals, inteda it uses typed id
bb2ac037031beb4	2016-08-05 13:30:51	PFCD-2252	PFCD-2252 User group grid filters do not work for multiple selections
1db8c986c232fd8	2016-08-10 09:44:10	PFCD-2254	PFCD-2254 Add missing image
929917edfa9cc6d	2016-08-10 15:34:37	PFCD-2258	PFCD-2258 Grouping in PA Data Tables some times does not work
beb02aa1093e203	2016-09-15 10:24:24	PFCD-2268	PFCD-2268 Filter does to work on pp with datasource restricted columns
e30d97999ebd637	2016-09-02 09:59:29	PFCD-2269	PFCD-2269 avoid emptying outputs
4e5143e87cb09b1	2016-09-02 09:59:18	PFCD-2271	PFCD-2271 missing link to contracts in My Pending Approvals panel
a0bdda1773a9dc4	2016-10-03 18:25:14	PFCD-2318	PFCD-2318 - PX mass update/delete authorisation error
664832e6221d70f	2016-09-09 16:28:44	PFCD-2330	PFCD-2330 enable overrides in Contracts
69c09b69b8961b6	2016-09-16 07:32:00	PFCD-2336	PFCD-2336 label for productPicker input
559c3b16c65cf10	2016-10-07 12:01:57	PFCD-2349	PFCD-2349 workaround for sgwt bug in new filter buttons
e001e7f5e8c7bbf	2016-10-23 15:57:56	PFCD-2438	PFCD-2438 - Roles Product/Customer Management (Read-only) - PX/CX access missing Role "Product Manager (Read Only)" should have a read-only access to Product Extensions. Role "Customer Manager (Read Only)" should have a read-only access to Customer Extensions.
92d2baadcce5c5b	2016-11-11 11:16:48	PFCD-2441	PFCD-2441 P,C,PX,CX list Create Change Request button PFCD-2441 add crate dcr button to px and cx page
92d2baadcce5c5b	2016-11-11 11:16:48	PFCD-2441	PFCD-2441 P,C,PX,CX list Create Change Request button PFCD-2441 add crate dcr button to px and cx page

pricefx-web ChangeLog All Changes

Commit	Date	Message
92d2baadcce5c5b	2016-11-11 11:16:48	PFCD-2441 P,C,PX,CX list Create Change Request button PFCD-2441 add crate dcr button to px and cx page
e001e7f5e8c7bbf	2016-10-23 15:57:56	PFCD-2438 - Roles Product/Customer Management (Read-only) - PX/CX access missing Role "Product Manager (Read Only)" should have a read-only access to Product Extensions. Role "Customer Manager (Read Only)" should have a read-only access to Customer Extensions.
559c3b16c65cf10	2016-10-07 12:01:57	PFCD-2349 workaround for sgwt bug in new filter buttons
a0bdda1773a9dc4	2016-10-03 18:25:14	PFCD-2318 - PX mass update/delete authorisation error
0662fc1c48069f6	2016-10-03 13:17:15	Fix XSS issues after SFDC review
956c9038a9a0eb8	2016-09-30 16:40:59	Clear quote inputs more thoroughly to reflect changed inputs on response
103fbe0f4a4e2d9	2016-09-30 15:48:36	Clear quote inputs more thoroughly to reflect changed inputs on response
69c09b69b8961b6	2016-09-16 07:32:00	PFCD-2336 label for productPicker input
beb02aa1093e203	2016-09-15 10:24:24	PFCD-2268 Filter does to work on pp with datasource restricted columns
9c5adcb1a682cef	2016-09-11 20:49:05	PA sim fix: was unable to select regular (PB) formulas
761bd3d1acefb01	2016-09-11 20:42:55	PA sim fix: was unable to select regular (PB) formulas
c49e4809b952f15	2016-09-11 19:53:29	PA sim fix: was unable to select regular (PB) formulas
664832e6221d70f	2016-09-09 16:28:44	PFCD-2330 enable overrides in Contracts
e30d97999ebd637	2016-09-02 09:59:29	PFCD-2269 avoid emptying outputs
4e5143e87cb09b1	2016-09-02 09:59:18	PFCD-2271 missing link to contracts in My Pending Approvals panel
65445f6bd32c50e	2016-08-29 18:54:23	PFCD-2248 upgrade sgwt to 6.0p-2016-08-27
e829c3d06cb21d9	2016-08-26 11:29:01	PFCD-2248 - implementing spinner wheel and adding try catch block to ensure mask is turned down when task ended correctly or not
457068b05779548	2016-08-11 11:45:17	PFCD-2251 - upgrading to sgwt with fix
929917edfa9cc6d	2016-08-10 15:34:37	PFCD-2258 Grouping in PA Data Tables some times does not work
90deb6ff03ee2b0	2016-08-10 11:54:50	PFCD-2249 - set the icon size to the original one
659f549cba5f5ad	2016-08-10 11:54:50	PFCD-2249 Calendar buttons disappeared after PFCD-2247 The SGWT update to fix PFCD-2247 broke our EditableMiniDateRangeItem, a customized version of the default form item that shows up in the date cells of the filter row at the top of a grid.
5013f68e489025f	2016-08-10 11:54:50	PFCD-2251 4 more validators have same issue for "no dupe" implementation
18aa051a1f33ff9	2016-08-10 11:54:50	PFCD-2251 - changed validator to not use equals, inteda it uses typed id

6ff38ea7004c40e	2016-08-10 11:54:50	PFCD-2174 - fixes ambigous method call
71272b620b016c2	2016-08-10 11:54:50	PFCD-2247 Bump SmartGWT to build 6.0p-2016-08-04 Known regression linked to this commit: PFCD-2249
1db8c986c232fd8	2016-08-10 09:44:10	PFCD-2254 Add missing image
e35b7b3cca8534e	2016-08-08 11:21:22	minor refactoring
1deca8492fc05b0	2016-08-08 10:58:44	PFCD-2240: LPG: disable the row(s) and hide the relative approve/deny buttons until the approve or deny actions are in progress
024547e4d376e32	2016-08-05 15:08:00	[EC] Upgrade to Excel-DNA version 0.33.9
bb2ac037031beb4	2016-08-05 13:30:51	PFCD-2252 User group grid filters do not work for multiple selections
0e3739bad9b9c7b	2016-08-03 08:25:20	PFCD-2244 Add "part_no" option into PriceAPI
463cfff4bcbf782	2016-08-02 11:35:39	PFCD-2239 - initial mask is skipped - it is distracting
a95686f1d9e0eb6	2016-08-02 07:20:38	PFCD-2239 - sonar fixes
61affbcf6360952	2016-08-02 07:10:57	PFCD-2239 - fixing tabs and added modal mask also for autofit
7b7f217e2e75f32	2016-07-29 07:08:36	PFCD-2236 - Show/hide column / apply prefferences / reorder should be faster and show modal mask when applied -> mask is shown when changing layout, only for LPG now -> render grid is not used anymore -> creted custom vibility menu - used for custom event that properly cleares caches
093233c61084e01	2016-07-28 12:15:00	PFCD-2108 Paramater details are not imported when using import/export groovy
87405650ec2af8d	2016-07-27 13:51:03	PFCD-2229 Dashboard controller doesn't open deep links (typedId)
32938aacb1b09b9	2016-07-27 06:22:34	Make direct link fleld in start page panels non-sortable and non-filterable
4d21a69047a8e35	2016-07-25 08:15:06	PFCD-1657 Make the approval email's subject customizable
0d6287035a98e0f	2016-07-25 07:14:08	PFCD-2222 Special role "User group override" only
a209e88681bcab8	2016-07-19 16:31:52	PFCD-2215 - implemented cache eviction and full row render
749573fa4c9109a	2016-07-19 11:32:05	PFCD-2172 Also apply in-flight/pending input value changes when hitting "price quote" (i.e. complete edit before sending quote object) Backing out approach again
02e1e2108668786	2016-07-19 11:11:15	PFCD-2172 Also apply in-flight/pending input value changes when hitting "price quote" (i.e. complete edit before sending quote object) Missing import
b58c512005754c7	2016-07-19 11:07:31	PFCD-2172 Also apply in-flight/pending input value changes when hitting "price quote" (i.e. complete edit before sending quote object)
6346cdc973d7f15	2016-07-19 10:59:01	PFCD-2001 don't fire showData in prefs after callback on console init
63c2070be1983ad	2016-07-19 07:29:42	PFCD-2172 Also apply in-flight/pending input value changes when hitting "price quote" (i.e. complete edit before sending quote object)
a272f576ed2bdb2	2016-07-18 18:11:40	PFCD-2151 Update filter controller even when filter set to null Also added one more FilterController.setFilter() method to keep the controller's link title consistent between its initial title, its title after reset, and its title when set to null.

e1801e3761165f9	2016-07-18 18:05:48	PFCD-2151 Make sure to save a Data Load's filter when it's set to null
a54d460766f8b8f	2016-07-15 07:44:14	PFCD-2194 Add download button to security/login trail logs
d56d174066b6e28	2016-07-14 12:18:31	PFCD-2197 fix ResultMatrix in single dashboard layout Beware, this single dashboard layout steals state from its inner portlets!
138959399e3c36b	2016-07-14 11:13:59	PFCD-2191 - Gauge cut off in QC UI
81fd3fec6f5564a	2016-07-13 07:44:19	PFCD-2152 Fix a regression with Data Feed and Datamart data tables For users without the SchemaEditor role, the data tables were not displayed anymore.
d1550ceea4217f8	2016-07-13 07:39:49	PFCD-2152 show only relevant action buttons in DF, DS and DM lists Delete button was still showing up sometimes, if a user without SchemaEditor role selected a Data Source
292f8b04591a37b	2016-07-12 13:04:30	PFCD-1413 Don't allow "limit to" filter for Boxplots For charts that support embedded Boxplots, the "limit to" and "boxplot" options are now mutually exclusive.
e6c7885ff4e75ac	2016-07-11 12:07:56	PFCD-2152 show only relevant action buttons in DF, DS and DM lists Show/hide the "add", "delete", and "import/export" action buttons in the left hand side lists of DFs, DSes, DMs, and SimDMs according to user's role and to current selection.
d53327d7f6185c8	2016-07-08 11:08:01	PFCD-2146 update prefs after save
71ce4e796eeff63	2016-07-08 11:07:45	PFCD-2146 invalidate prefs cache on prefs save in dashboards
6716f65e609052f	2016-07-08 11:07:33	PFCD-2146 delete pref button disabled after prefrences successfuly saved
ca79de1e58f8720	2016-07-08 11:07:18	PFCD-2123 minimized saved portlets are loaded not minimized on dashboards
03baecfa3b5e84b	2016-07-08 11:06:57	PFCD-2105 fix ResultMatrix filtering and make sure refresh is ok in dashboards
502a2de7b31d323	2016-07-08 11:06:41	PFCD-2146 single preferences and dashboards
f9ba452a38470e5	2016-07-07 08:37:31	Updating system command view
497478f229046a1	2016-07-07 07:10:25	PFCD-2094 Make list grid item adjust width
1ac1eceddb8dc51	2016-07-06 09:23:08	PFCD-2152 PA: Hide deploy button + lock field editor if user not admin
4bebddc5fada556	2016-07-01 13:15:55	PFCD-2154 PA Data Table group labels sometimes disappearing
05e1b451b37bf27	2016-07-01 13:15:55	PFCD-1301 Custom number format in PA data grids not applied as expected Was done according to column number, so applying wrong format if user moved columns around.
df8d356e0e045d9	2016-06-30 08:23:25	PFCD-2143 Add Pricelist import/export/archiving option
d5558f188fb0abb	2016-06-29 14:34:58	PFCD-1526 Open multiple PX/CX tabs
7d6e93fe6ed3ce6	2016-06-29 09:12:14	PFCD-2139 do make start page grids selectable
d176c10c0eaaf92	2016-06-23 11:40:52	PFCD-2109 Simple and advanced filters sometimes active at the same Sometimes the AdvancedFilterBuilder and the basic GWT filter row at the top of the grid could be displayed simultaneously. Was due to the fact that after filterBuilder.setVisible(true), filterBuilder.isVisible() could still return false if the grid had not been drawn yet.

57c6bfc10fe5930	2016-06-23 07:03:45	PFCD-2124 MultiTierEntry hint has huge width
ad660bb576b5140	2016-06-17 14:06:30	PFCD-2118 Show edit details button also for product details -only editors
c24842bff8dd2e4	2016-06-16 15:03:03	[EC] Fix for fetchURL meta entries missing the leading '/' char
e46342ad52d6e8e	2016-06-16 14:40:44	PFCD-2112 Remove useless copy action from Rebate Records grid
9eb07c9a63eda67	2016-06-16 08:40:40	PFCD-2110 do not ever hide first multitier input param
fd5c1d8cd52a7a0	2016-06-15 07:46:08	PFCD-2054 Hide action buttons of DM DataEditor until a source is set
32f3ac6b771a7bb	2016-06-13 16:08:03	PFCD-2080: Ability to filter on fields with 'Expression' : calc fields do not show up in FilterBuilder
6f9124dcb6d354a	2016-06-10 07:31:30	PFCD-2063 Show/hide LPG approve/deny buttons in a more consistent way The same logic is now used to decide whether approve/deny buttons should appear on the row of an LPG item, and at the bottom of the grid if that same row is selected.
b8981fc73b5b0d9	2016-06-09 23:53:00	Revert "PA chart calcWidget to use query formulas, not data load formulas" This reverts commit 8b046030c9d9051f74d004e37e75962fbf30b4af.
8b046030c9d9051	2016-06-09 23:26:00	PA chart calcWidget to use query formulas, not data load formulas
b6553f23f99f914	2016-06-07 14:03:58	PFCD-2100: PA: init/row/summary group options missing in Query Calculation Logic (formula editor)
7108ebbc9b650d0	2016-06-02 21:11:15	PFCD-2037: fixing 'Rebate Text' widget, now showing and allowing to change RR.description
05f0d4a9c261998	2016-06-02 12:32:13	PFCD-2070: PA: do not show 'PerUnit' & 'Percentage' aggregation functions in Chart controller if no basis field set in the DM
8e88479c05e15d3	2016-06-02 12:23:51	Resolving Sonar warnings....
71ba21bf8b8ae64	2016-06-01 14:45:32	PFCD-2048 AdvancedGrid delete button visible only if a row is selected This is usually handled in classes using or extending Advanced(List)Grid but it was still missing in so many places that it is now handled at the AdvancedGrid level.
3ccf555f44a6672	2016-05-31 15:35:08	PFCD-2055: Copy of scheduled job leads to an error
dcacedba713390e	2016-05-30 13:49:48	PFCD-2077 Configurator for Logic doesn't work correctly -> editorExitHandler is ommitged for matrix widget.
3d75055a58bd64f	2016-05-30 12:31:00	PFCD-2080: Ability to filter on fields with 'Expression' PFCD-1807: Datasource UI - editing manually record results in duplicate row

Best Practices

The following articles provide useful tips how to boost performance and stability on your projects and write logics effectively.

- Performance Practices
- Functional Practices
- Coding Style
- Naming Conventions
- Backend Error Messages
- Customer Volume Metrics

Performance Practices

- Logic Structure
- Data Querying - api.find()
- Avoid Big Data in Element Result

Logic Structure

Based on experience, it turned out that there are different element types that follow certain logical structure. This table shows how to organize those elements into corresponding blocks based on their logical order.

Element Block	Element Name	Sample code	Display	Comment
Retain global element	**RetainGlobal**	`api.retainGlobal = true`	None	If api.global value should be kept for every item. ✓ From 3.8 Manhattan, api.retainGlobal can be set to true by default in Configuration. Not needed for dashboards, since they calculate a single item.

Gathering data
need for
parameters
gathering
	SalesOrgs	`if (!api.global.salesOrgs) {`
		`api.global.salesOrgs =`
		`api.findLookupTableValues("SalesOrg")`
		`.collect {`
		`it.name`
		`}`
		`}`
		`return api.global.salesOrgs`
			None	Only queries really needed for parameters gathering should be placed in this block.
				Keeping each data query in a separate element avoids element timeout.
				Make sure the display is None, so this value does not get persisted.

Parameters
gathering
elements
	SalesOrg	`api.option("SalesOrg",`
		`api.getElement("SalesOrgs"))`
				Keeping each user entry / input in a separate element is useful when testing the logic.

	Currency	`api.option("Currency")`		
	...	`...`		

Stopping
syntax check
execution
element
	SyntaxCheckAbort	`if (api.isSyntaxCheck()) {`
		`api.abortCalculation()`
		`}`
			None	Cancels the run if running in the syntax check mode (parameters gathering).

Marked item
dirty check
element
	ReferencePrice	`...`
		`if (needsSecondPass) {`
		`api.markItemDirty()`
		`//api.abortCalculation() - to be considered`
		`}`
		`...`
				It is good to check if the item needs a next pass at the very beginning of the logic if you plan to call api.abortCalculation.

Data gathering
for logic
calculation
	CostRecord	`def filters = [`
		`Filter.equal("Cost"),`
		`...`
		`]`
		
		`def pxCosts = api.find("PX", *filters)`
		
		`if (pxCosts) {`
		`return pxCosts.getAt(0)`
		`}`
		
		`return pxCosts`
			None	Returns the PX Cost record at one place, since other elements can refer to it.
				Make sure the display is None, so this value does not get persisted.
				Keeping query in a separate element is useful when testing the logic.
				The data querying should be done using api.batchInfo.

	Margin	`return api.vLookup("Margins", ...)`		
	MinMargin			
	...	`...`		

Calculation
elements
	CostPlus	`def cost = api.getElement("Cost")`
		`def margin = api.getElement("Margin")`
		
		`if (cost != null && margin != null) {`
		`return cost * (1 + margin)`
		`}`
		
		`return null`
				Calculates all the elements using the data queried in the previous block.

	...	...		
Result price element	**ResultPrice**	`return api.getElement("CostPlus")`		Calculates the result price.

Auto-approve
element
	AutoApprove	`def isAutoApproved = ...`
		
		`return isAutoApproved ? 1 : 0`
				When calculating LPGs with auto approval functionality.

Data Querying - api.find()

- Always use a maxRow limit
- Specify subset of fields to be retrieved
- Avoid multiple occurrences of the same api.find()
- Use of aggregation functions
- Use filter for name field when querying PX or CX

Always use a `maxRow` limit

The api.find() function without the `maxRow` parameter specified returns only the first 200 records by default. That often causes a problem because if you don't analyse how many records you can potentially get, you may miss some data from your result. So that is the reason why you **should always specify the `maxRow` parameter**. The maximum value you can set this to is `api.getMaxFindResultsLimit()` which by default gives 2000 records (the value can be increased for customer dedicated instances).

If you still need to retrieve more data than `api.getMaxFindResultsLimit()`, you will need to use `api.stream`. In such a case, make sure that you always call `close()` on the iterator.

✓ From 3.8 Manhattan, you get a warning if the number of rows returned equals 200 or formulaEngine.script.findMaxResults and the maxRows parameter of the find was not specified (=0).

Specify subset of fields to be retrieved

The call of `api.find()`, without the `fields` parameter specified, retrieves all the fields of the given domain object from the database and transfers that data over the network to a given calculation node. But very often you only need only a few fields to work with and retrieving all the fields causes unnecessary data transfer. So if you are querying e.g. a product master (that has 43 fields) and you use only 4 fields in the logic, the data transfer over the network is 10 times higher than needed. Therefore you **should specify the list of fields in the `fields` parameter**.

TypeCode	Number of fields
P	43

PX20	28
PX30	38
CX30	38
PGI	146
PR	51

Avoid multiple occurrences of the same api.find()

The function `api.find()` is heavily used in calculation logics and each call can be quite costly. As the logic evolves during the time and various team members need to query the same data, the common mistake is that they create the same or similar `api.find()` query.

That increases the query execution time twice, so therefore you should call `api.find("PX")` only once in one of the very first elements of the logic and mark it as Display: None (not to have the records persisted in the database) and then refer to the results using api.getElement(). Alternatively, you place the common api.find() searches to the Groovy library as a separate function (e.g. like Queries.findStandardCostRecord(...)).

It is definitely **not** a good practice to copy & paste api.finds.

Use of aggregation functions

Consider using the aggregation functions SUM, AVG, MIN, MAX in `api.find()`. So instead of iterating all the records, use the following special Map construct to define the aggregation:

```
api.find("PX", 0, 1, null, ["attribute1": "SUM"], true, filters)
```

Use filter for `name` field when querying PX or CX

Since the PX or CX tables are virtual tables and are stored in one physical table (depending on a type, e.g. PX20 or CX20), you always have to specify the filter for the `name` field, because two virtual PX tables can be stored in different physical tables. This will instruct the engine which physical table to look (e.g. PX10 or PX20) at.

Avoid Big Data in Element Result

If an element returns big amount of data in the result, then persisting such result in the database can cause serious problems.

Compared to Java, Groovy does not force you to use return statements. However, this is not a recommended practice, since it can cause unexpected results. Imagine a situation that you do not need the element to return a value (e.g. because you only want to set api.global) and at the end of an element you leave an expression that is evaluated as a big list or map.

The best practice is to use return statement at the end of each element.

Also, if you return e.g. a List or Map in some element as a temporary result, that element should have the Display option set to Never.

Recommened way

```
//calculate result somehow
return result
```

Avoid Re-calculations

Calculate a flag

Sometimes the logic calculates over and over the records that are not changing (CFS, Datamart). If a record needs to be calculated only once, it should be marked with a flag and the calculation should exclude records with that flag. Further consider performing incremental updates in

Datamarts.

Multiple passes

If you call `api.markItemDirty()` on an item, it is generally good to to in the beginning of the logic and also call `api.abortCalculation()` afterwards.

api.retainGlobal

If `api.global` is used, `api.retainGlobal=true` must be the first command of the first element called "GlobalCache" of the logic (even before the library elements).

A common problem is that setting `api.retainGlobal` tends to be forgotten due to a human error. Because of that, the whole `api.global` cache gets lost between the calculation of the records; instead the whole cache is calculated for each and every record again and again. It is very difficult to discover this kind of problem later on and determine if `api.retainGlobal` was not set in an existing project on purpose or by mistake.

The function `api.retainGlobal` cannot be deprecated due to a backward compatibility.

ℹ See also api.global versus api.local for additional information.

✓ From 3.8 Manhattan, api.retainGlobal can be set to true by default in Configuration.

Product Extensions

Suitable PX Size

Initially, all the PX tables were stored in 1 physical table in MySQL and the database engine was not performing well with big amounts of records. As the data was growing, `api.finds("PX")` became slower. Also the customer was getting timeouts in the UI. Database operations (like data correction) were almost impossible. Therefore the PX tables were split into several physical database tables depending on the number of columns (PX3, PX6, PX8, PX10, PX20, PX30, PX50). Choosing the right PX size helps you to reduce the size of the physical table, so you should really use the appropriate size and not e.g. PX30 for everything.

Database Indexes

Database (DB) indexes significantly impact querying the database data. Using the right index is a key factor when working with product extensions (PX) with hundred thousands of records or more. Pricefx stores multiple PX tables of the same column size in one physical database table (PX3, PX6, PX8, PX10, PX20, PX30, PX50). Assume that there is a PX "Stock" that requires – for best performance – to have a DB index defined as (attribute5, attribute2) and another PX "SalesPrice" would require index (attribute2, attribute3). And other PXs might require different index every time.

The solution is to define a single index e.g. (attribute1, attribute2, attribute3) on the physical table which will be shared by all PXs. It means that you configure the most significant fields of the PX as attributes 1, 2, 3. The significant fields for each PX are the ones mentioned in the filter of `api.find("PX")` in the logic that you want to speed up.

Logic Element Timeout Override

Within a logic it is possible to override the timeout of a particular element.

While this can be useful in some cases (or even required), this may prove to be a way to actually hide a performance problem.

In one of the example cases, the timeout override was increased to value 900. After careful analysis of the logic element Groovy code, we identified that a change there brought back the execution time to less than 30 secs!

The performance stack can be found in the UI for non-distributed jobs. For distributed jobs, this is trickier and you can exceptionally ask a Support engineer to provide this to you.

Carefully analyze the performance stack to understand what portion of the code takes long and address that with a code or design change.

> To summarize it, setting a timeout override to a large value (e.g. 900 seconds) generally hides a performance problem.

Performance Improvements (Batching, api.getBatchInfo)

Overview

This methodology is based on smart usage of api.global – together with api.batchInfo() we can make use of it to speed up the process of fetching data from PA and different sources and to save the processing time and hardware usage (especially limit the PA queries or api.find/stream up to 200 times).

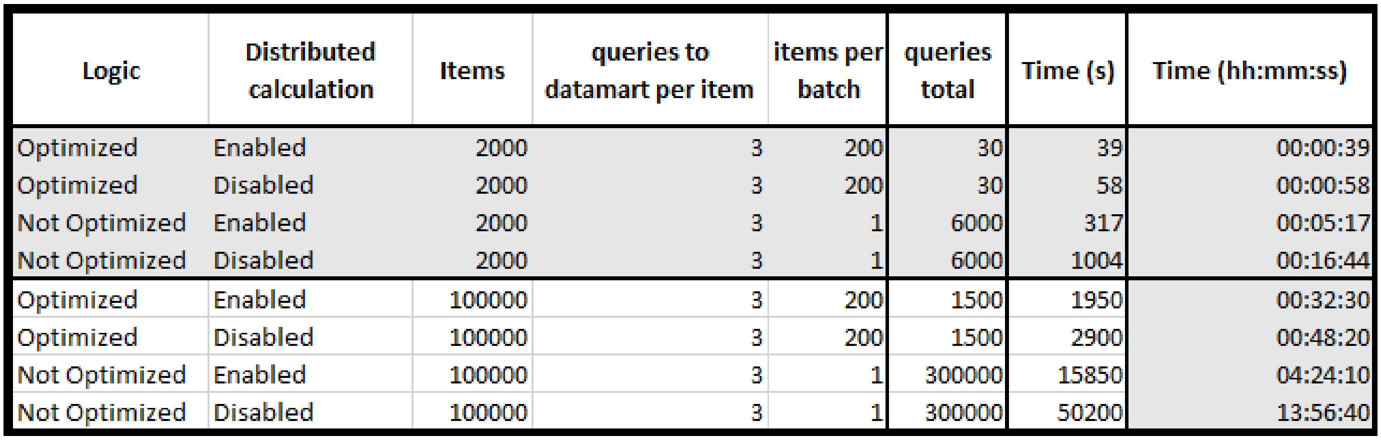
api.global is thread safe which means that the cache can be shared as long as the different parts of the calculation are calculated on the same thread. For example when you run a job with 'Allow Distributed Calculation' unchecked, there will be only one node and one thread meaning that the global cache can be shared for all of the items in calculation (not just batch). When 'Allow Distributed Calculation' is checked, there can be several nodes taking care of the calculation (which will have several threads with different batches, meaning that the cache is shared for one thread only). But when we have a good implementation of the code, we can speed up the process using 'Allowed Distributed Calculation' together with api.global cache.

api.getBatchInfo() shows the items that were added to the batch, so you can see which items are going to be calculated next. It shows results up to 200 items or NULL while in the debug node (so it is recommended to handle this case for development too). It is always a list of two dimensional arrays, PL/LPG context [[sku,secondKey]....] , in CFS [[typeCode, secondaryKey]]. When using it in CFS, you have to convert typeCode to Product SKU / Customer ID or any other unique ID to get the correct data.

Differences in distributed calculation were mentioned there (including batching flow example): Distributed Calculation

MEASURES

Why is it so important? Let's check an example of processing times.



Logic	Distributed calculation	Items	queries to datamart per item	items per batch	queries total	Time (s)	Time (hh:mm:ss)
Optimized	Enabled	2000	3	200	30	39	00:00:39
Optimized	Disabled	2000	3	200	30	58	00:00:58
Not Optimized	Enabled	2000	3	1	6000	317	00:05:17
Not Optimized	Disabled	2000	3	1	6000	1004	00:16:44
Optimized	Enabled	100000	3	200	1500	1950	00:32:30
Optimized	Disabled	100000	3	200	1500	2900	00:48:20
Not Optimized	Enabled	100000	3	1	300000	15850	04:24:10
Not Optimized	Disabled	100000	3	1	300000	50200	13:56:40

This is just to give you an idea what it means to optimize results for simple 3 queries and 2000 and 100 000 items calculation.

Notes:

- The real numbers are greyed out, done on the 2000 level.
- The 100 000 items is a projection, but the results will be much better than shown due to the fact that distributed calculation will send the work to more than 2 nodes (as it was for 2000 items).
- Optimized = querying using api.getBatchInfo() together with api.global cache following current guide.
- Not optimized = querying on item level (1 to 1) like in most projects.
- Distributed = distributed jobs of calculations (send to x nodes). Allowed Distributed Calculation = Checked.
- Not distributed = single node & thread work. Allowed Distributed Calculation = Unchecked.

Implementation (LIB Example)

Overview of the lib:

Common library: https://bitbucket.org/pricefx/pricefx-logic

Particular example: https://bitbucket.org/pricefx/pricefx-logic/src/BatchUtils/CalculationLogic/CommonLib/elements/BatchUtils.groovy

Prerequisites:

Methods of the presented library (as of today):

- libs.CommonLib.BatchUtils.prepareBatch(sku) – This method initializes the cache variables that can be used later on (should be called right **after** api.retainGlobal = true in the logic with feed of sku).
    - Will create a new local reference api.local.isNewBatch – boolean value (true/false) shows if you are working with a new batch or not.
    - Will create/override a new global reference api.global.iterationNumber – shows if this is 2nd pass or 3rd pass of calculation etc.
    - Will create/override a new global reference api.global.currentBatch as Set of the given key.
- libs.CommonLib.BatchUtils.isNewBatch() – returns a boolean value showing if data should be fetched and cached from the sources or not.
- libs.CommonLib.BatchUtils.getCurrentBatchSku() – returns the Set of elements that should be fetched and cached later on.

Implementation:

1. You need a LIB, which is located at: https://bitbucket.org/pricefx/pricefx-logic/src/master/

2. You need to apply it as required. Here is an example of fetching PX data and caching it for the batches using api.stream():

Note: The presented LIB is under development, currently it does not support **CFS** and **2nd key logics (MATRIX)**.

Implementation (NON-LIB example)

The implementation may vary but the idea should be similar. This example is from one of the projects for PL/LPG calculation (as for CFS the api.getBatchInfo() sends typeCodeID, not the SKUs).

In the beginning we need to override retainGlobal using:

Then as an example, make sure that the api.global.batch is initialized while it does not exist for the current thread.

In the next step, check whether the SKU is already cached or not. More details are in the comments in the code.

To use the cached cost in a different element, you can access it like this:

This was a simple example of usage in api.find. When the given cache should be shared for other 199 items without the need to access the PX table again, you can extend this functionality. There is another example within a PA query:

The given examples were for api.find and Datamart queries, however you can use them in any other strategy, including pre-calculations of data, fetching a logic for a given SKU, playing with the 2nd key etc.

Warning

- When you work on a small amount of data, api.getBatchInfo() returns NULL. Using a shared library will solve this issue and it will always take at least one item. (In the pricefx-server settings, you can set when a batch is going to be created. I tested on 2000+ items and there were no issues, but less **could** return a value of NULL.)
- api.find() has a limited number of rows that you can fetch (you can use api.getMaxFindResultsLimit() to get the limit for the given environment and partition). Make sure that the data is not cut in the middle of the query or handle it with startRow/maxRows. As an alternative, you can use api.stream to cache the data and then work on it, especially while playing with the 2nd key.
- Datamart queries have limitations for the data that can be loaded (it is not exposed in the UI and it is cluster wide as pricefx-server settings). As an alternative, use streams.
- 2nd and 3rd... passes will create new batches, so the data will be loaded and processed again.
- Make sure that you set the element timeout for a sufficient period of time, as in the logic you will be fetching data for up to 200 SKUs and the query result time will be higher than for a single SKU.
- Find and others functions in PFX have their own limitations in terms of maximum rows that will be represented to the user. Make sure your logic and data do not exceed those limits; if they do, use streams or ask Support to extend the limits. But a good implementation of the code and good configuration (limitations, requirements, business keys of PX, filters on the different context) will almost always solve the issue.
- Once there was the following issue: Allowed Distributed Calculation NODE1 was fetching values correctly from PA, however NODE3 was not. The issue was resolved later on, but it was hard to troubleshoot, so it is good to have a few log outputs for such cases.
- If you need to store the data for all nodes in the calculation flow (in Distributed Mode), do not use api.global as it will be different for all nodes and threads; use a shared cache instead.

Performance Impact of Using api.currentContext() vs. api.currentItem()

If you ONLY need to get access to the current item being processed in your list of items (i.e. PL or PG), then you should use the 'api.currentItem()'.

If you need access to another item in your list of items, then you should use 'api.currentContext()'.

Performance wise there is a major difference as the 'api.currentContext()' retrieves the full list of meta data information too. So for example, if you are processing a list of 1M items and each item has 100 columns, this will trigger 100M metadata searches. This can be time consuming.

Performance Issues Inside Loops or Closures

- Looping through hashmap
- Caching the same values

Looping through hashmap

Looping through a hashmap can cost a lot of time. One example to illustrate it:

```
 9962.25 58% 46  CleanupPXes
18.920% 73   search(ProductExtensionAttributeMeta)
60.780% 46   find(PX3)
60.630% 46    search(PX3)
61.600% 46   find(PDESC)
 3.360% 46    search(Partition)
57.890% 46    search(ProductDescription)
27.650% 14   find(PX6)
27.600% 14    search(PX6)
41.930%  6   find(PX20)
41.910%  6    search(PX20)
 133.821%  6   find(PX8)
 133.791%  6   search(PX8)
```

```
def actSalesPrice = pxSalesPriceMap.findAll { it.value.attribute1 ==
actOutlet }
```

This has cost 3 ms x 268 to be run which means almost 1 second for this element.

To improve this, prepare another object – a hashmap or list through which your engine can go much faster.

```
def actSalesPriceImproved = pxSalesPriceMapImproved.get(actOutlet)
```

Caching the same values

Instead of calling api.find everytime when needed it is better to cash it into api.local / api.global.

Cashing makes even more sense across more items.

Too Many PA Queries

Too many PriceAnalyzer queries throttle the server.

In other words, we allow a maximum of 600 PA requests per minute on PROD instances (60 PA requests per minute on QA boxes) for a given job (for example a CFS or LPG). For every occurrence where a logic goes beyond this limit, the server will make it wait for 1 second.

This can have a very important impact on the overall performance of the logic.

A common error is to use the PA tables for storing non-PA data.

Another way to address this is to query the PA data for the current batch of SKUs being processed as opposed to one SKU at a time. For more details see Performance Improvements (Batching, api.getBatchInfo).

Avoid DB Queries Containing IN Clause with Thousands of Values

There are many ways to make a database "unusable".

One of them is using IN clauses in queries. While this can be very useful, it can also have a very adverse performance effect. Indeed, when you have thousands of values in your IN clause, the database often ends up having to do full scans. This will use heavy resources on the database side.

Avoid this scenario where possible.

Default Sorting Preferences for Price List or Price Grid Objects

For the purpose of this explanation, I will refer ONLY to price lists, but the same applies to price grids.

When you create a new price list and open it for the first time, you may see that no preferences are applied. Once it is open, you can create and apply a preference. But the initial opening of the price list may be slow. This is because the initial opening will sort the price list items by their system ID. And based on the default indexing scheme this may be slow.

To address this, you should create a default preference and have it applied to all price lists upon initial opening. This is done in the Configuration s ection. Generally you sort by the SKU or a combination of SKU and key2 for the matrix price lists.

Try out the sorting to ensure that this has the expected performance before applying as a default.

Bulk Load Price Lists

There is a server side setting which can impact the speed at which price lists get processed:

```
<backgroundWorkers>
    <bulkloadPricelists>true</bulkloadPricelists>
</backgroundWorkers>
```

This setting can help in the case of large price lists for which the processing logic is extremely fast. It impacts the way the calculated data finally gets written back to the database. By default this setting is set to false.

Workflow Post Step Logic with LPGs and Performance Issues

When the workflow post step logic is used after the LPG calculation is done and auto-approved (or manually approved by the user for each item), it can cause issues. This post step logic runs only on the web/central node, while the LPG calculation runs on a different node. The problem is that while the post step logic runs for every item, the other node needs to wait. This can lead to serious performance troubles, especially if the post step logic is time consuming.

String Concatenation

String#concat is the fastest method for String concatenation.

https://groovy-programming.com/post/5732276432

api.abortCalculation() to Be Followed By 'return' Statement

The function call 'api.abortCalculation()' instructs the server to stop processing **after** the end of the current logic element.

If you want the logic to stop immediately after the 'api.abortCalculation()' call, then you need to introduce a 'return' statement.

Functional Practices

- api.global versus api.local
- Prevent api.isSyntaxCheck Issues
- Use GIT
- Error Handling (DRAFT)
- Avoid Copy & Paste
- Expressions in Groovy
- Data Update in Distributed Mode
- Data Update in Distributed Mode (Batching, api.getBatchInfo)
- Datamart Setup
- Security Setup
- Check Inputs (From User and Data Tables)
- Distributed Calculation

api.global versus api.local

INTRODUCTION

The `api.global` and `api.local` are a maps which are populated in the memory of each server node. Variable `api.local` is initialized to an empty Map before calculation of every item while `api.global` is (if the `api.retainGlobal` variable is set to `true`) initialized to an empty Map before the calculation of the first item. If the `api.retainGlobal` variable is not set, then `api.global` behaves the same way as `api.local` and therefore it is deprecated practice and `api.local` should be used.

✓ From 3.8 Manhattan, api.retainGlobal can be set to true by default in Configuration.

> If any of the variables `api.global` or `api.local` holds a lot of data, these variables should not be returned as an element result since the data gets usually persisted for each calculated record.

api.global

Remember that for distributed calculations `api.global` is populated on each server node and each thread independently. So if you store there the results of some data queries, those queries will get executed on every node (and not once as one may think).

You should always:

- Cache some reasonable amount of data to be used for multiple items. Storing big data causes bad performance when recalculating a single item.
- Define a constant to be used in many elements of the logic.

When `api.global` should not be used:

- For passing values to functions. These functions would later become difficult to use. Consider passing values to a function as function parameters.

> Whenever you want to use `api.global`, do not forget to set the `api.retainGlobal` variable to `true` in the first element of the logic; otherwise the value of `api.global` will get lost. This is a common error that causes performance issues when values which should be cached for all items are calculated for every item.

See also api.retainGlobal for details.

api.local

> Before using `api.local`, consider creating a new element (possibly with the Display option = None) and store the value there. Performance-wise it is identical since both `api.getElement` and `api.local` are represented by HashMaps. You will also debug the value more easily: as elements result(s) of a logic test. You also will not need to call `api.trace` to debug the value. Last but not least, it will also help you avoid potential naming clash of the `api.local` key.

So `api.local` should be used only in the following scenarios:

- If you set `api.local` in some element and you then need to change the value in some other element (try to avoid it though).
- You need to store multiple calculated values and you do not want to return Maps as an element result. For example, you calculate a price at certain place and want to set a price reason text at once. In that case, the element typically returns the price and sets the price reason as e.g. `api.local.reason`.

When `api.local` should not be used:

- For passing values to functions. These functions would later become difficult to use. Consider passing values to a function as function parameters.

Prevent api.isSyntaxCheck Issues

The syntax check mode was introduced to enable populating logic parameters. This is specially useful for dashboards. However, it is often a pain to adjust the logic after it is developed and starts working with a calculable record. In such cases, it passes the "syntax check" which does not take any calculable record into consideration and fakes the values.

To avoid issues related to parameters generation, I place the following construct right after populating the parameters (as close as possible):

AbortCalculation element

```
api.userEntry(...)

api.option(...)

if (api.isSyntaxCheck()) {
  api.abortCalculation()
}
```

Use GIT

It is recommended that:

- Everybody on the project uses GIT.
  - If the customer contributes on the project too, they must use it as well.
- Everybody commits their changes to GIT.
- Nobody does the changes directly at the partition (without committing to GIT as well).
- The actual version (of the configuration) from GIT can be deployed to the partition at any time.

Such approach helps avoid overrides and solve collisions.

Error Handling (DRAFT)

As of now, there are multiple approaches to handling errors:

api.throwException(String errorMessage)

Throws a calculation exception with the specified message and aborts the calculation.

api.addWarning(String message)

Adds a warning to the formula execution warning log.

api.criticalAlert(String msg), api.redAlert(String msg), api.yellowAlert(String msg)

Marks the current formula element with a critical alert and the given message.

Avoid Copy & Paste

Copy & paste is generally a very bad practice in software development. Avoid generating redundant code; rather create a Groovy library function.

Expressions in Groovy

All the operations like +, -, *, / etc. should be checked to not be null, e.g.:

```
if (revenue != null && cost != null) {
 def margin = revenue - cost}
```

In division, the divisor must be checked to be non zero, e.g.:

```
if (priceNew != null && priceNew) {
 def priceDiffPct = (priceNew - priceOld) / priceOld}
```

Data Update in Distributed Mode

This is currently under evaluation by the MediaSaturn team.

`api.addOrUpdate()` can only run in a non-distributed mode. A workaround is to use integrate HTTP transaction.

If this works, we will ask the ENG team to open this in Groovy API.

Data Update in Distributed Mode (Batching, api.getBatchInfo)

The idea behind this that for the Senetic project we had to create a logic which is less time and hardware consuming than the current solution.

In the past, the processing time for single `api.addOrUpdate` per product master item was 13–14 hours for 67 000 items, but in fact there were a lot of more items (in total 2 700 000 references) being added or updated in the PX tables since it has a business key on the SKU and domain.

A function was created to aggregate sales, promotions (Google adwords) and competitor data in one table with at least 3-4 queries to PriceAnalyzer, which in the past was an overkill at the shared production environment.

After applying the changes we ended up with **30 minutes** as the average processing time for the 67 000 items (and the 2 700 000 `api.addOrUpdate` calls).

api.getBatchInfo() does not return SKU or customer ID, it returns `typeCode.id` which has to be parsed to a specific source of your choice (in this example we use products).

Lib Example

To be done once the shared pricefx-logic lib is fully ready to cover the CFS part as well.

Non-lib Example

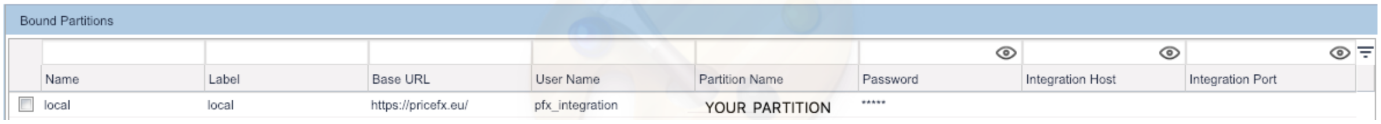
First, we collected all the items to the batch.

PARSE TYPECODE.ID TO PRODUCT SKU

SEND QUERY USING API.BOUNDCALL()

In this example we send the whole batch of items using API, not a single item (like in quotes or different modules where only a single item is shown; see Data Update in Distributed Mode (Single Item)).

Before you can use this function, you need to configure the bound call. If you have an integration behind, there is a high chance that it was used by the integration and so you can just refer to the configured one; if not, configure it using Bound Partition.



Bound Partitions

	Name	Label	Base URL	User Name	Partition Name	Password	Integration Host	Integration Port
□	local	local	https://pricefx.eu/	pfx_integration	YOUR PARTITION	*****		

In this case, we use a shared production base URL and an already configured pfx_integration user in the partition. We need to provide the user's credentials. A different user can be used too (your user or admin, even though I would not recommend using your user as it might be deactivated and the whole logic will break).

Datamart Setup

- Mark the **per unit basis** for the quantity/volume field.
- Distinguish if the data is **extended** or **per unit** amounts and mark them correctly.
- Mark the **percent basis** (recommended to do it based on a list price, not an invoice price as it treats on and off invoice discounts the same).
- Make sure the **data types** are correct.

Security Setup

The recommended practice is to start a project with the configuration business role and a customer business role and assign these roles to the users.

Check Inputs (From User and Data Tables)

It is good practice to check whether the data are complete and decide for an appropriate action.

- If configuration data is missing and calculation does not make sense without them, use api.throwException().
- If missing or wrong values are allowed, you can solve it by using alerts (yellow, red, critical) or use api.abortCalculation().
    - You can create an additional field which shows to the user the alert reasons.

Sample

```
def priceType = "CostPlus"

def price = null
def reason = "Unknown error"

def targetMargin = api.getElement("TargetMargin")

if (targetMargin) {
  def vat = api.getElement("VAT")
  def cost = api.getElement("Cost")

  if (cost && vat != null) {
      price = cost * targetMargin * (1 + vat)
      reason = "OK"
  } else {
    reason = "No Average or Calculated cost price in PX
Cost_RRP or VAT defined"
    api.redAlert(reason)
  }
} else {
  reason = "No target margin defined"
  api.redAlert(reason)
}

api.local.reason = reason

return price
```

Distributed Calculation

Distributed calculation is something that you can select in context of PL/LPG/CFS, which could speed up your processing time drastically, but how does it work?

The idea is pretty simple, as non-distributed calculation uses only **one** NODE and only **one** THREAD will be created, while the distributed one dynamically select number of NODES which has configurable numbers of THREADS that are going to be created per NODE to perform calculation.

Please look closer on provided examples of distributed types of calculations (enabled/disabled), as it was created without accessing the pricefx-core code, the idea should be kept as how does it work in real life.

- Non-distributed Calculation
- Distributed Calculation
- Q&A

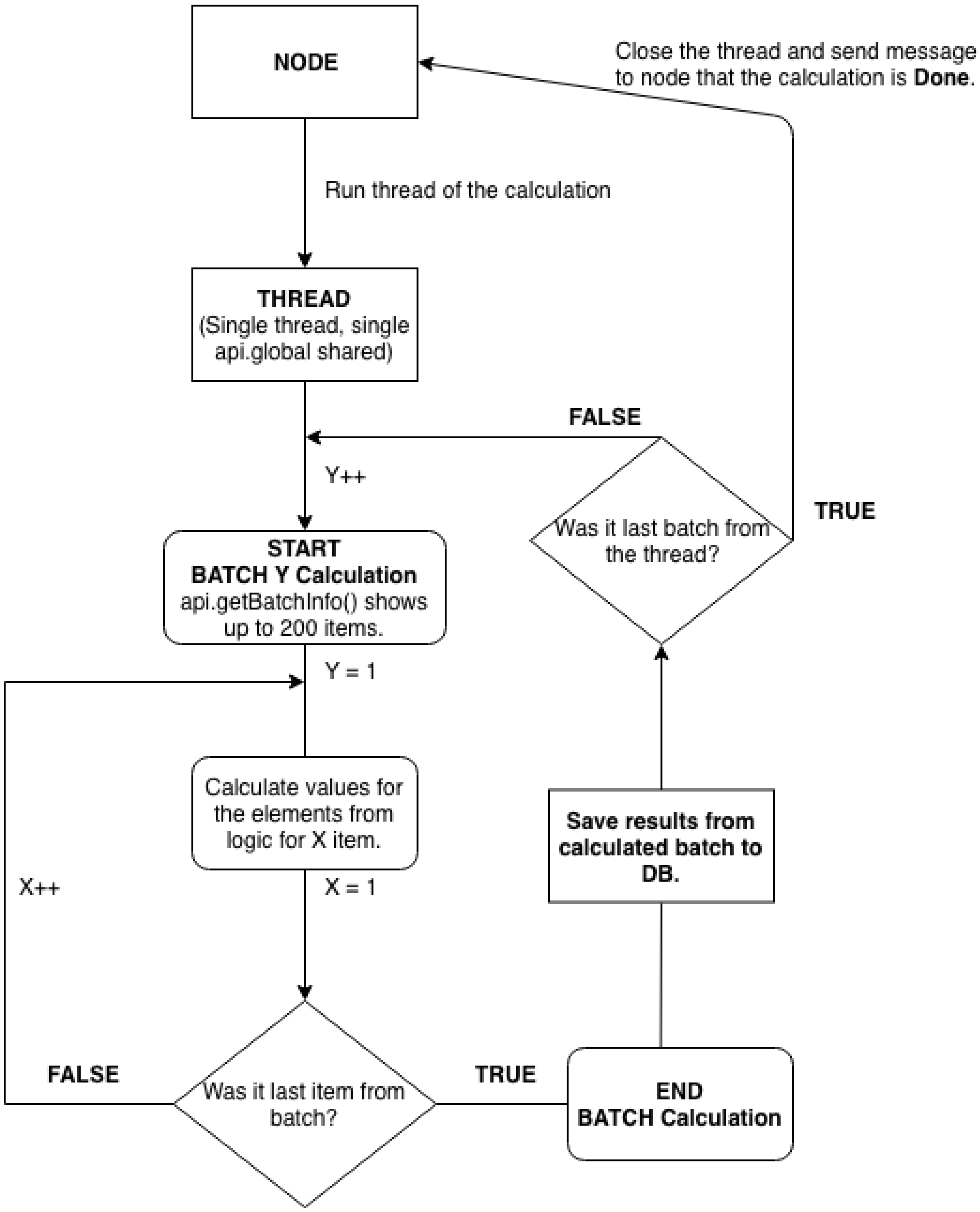
Non-distributed Calculation

This flow show example of calculation process for non distributed calculation.

Example:

☐ Allow distributed calculation

Allow Distributed Calculation (FALSE)


NODE
Close the thread and send message to node that the calculation is Done.
Run thread of the calculation
THREAD
(Single thread, single api.global shared)
FALSE
Y++
TRUE
Was it last batch from the thread?
START
BATCH Y Calculation
api.getBatchInfo() shows up to 200 items.
Y = 1
Calculate values for the elements from logic for X item.
Save results from calculated batch to DB.
X++
X = 1
FALSE
Was it last item from batch?
TRUE
END
BATCH Calculation

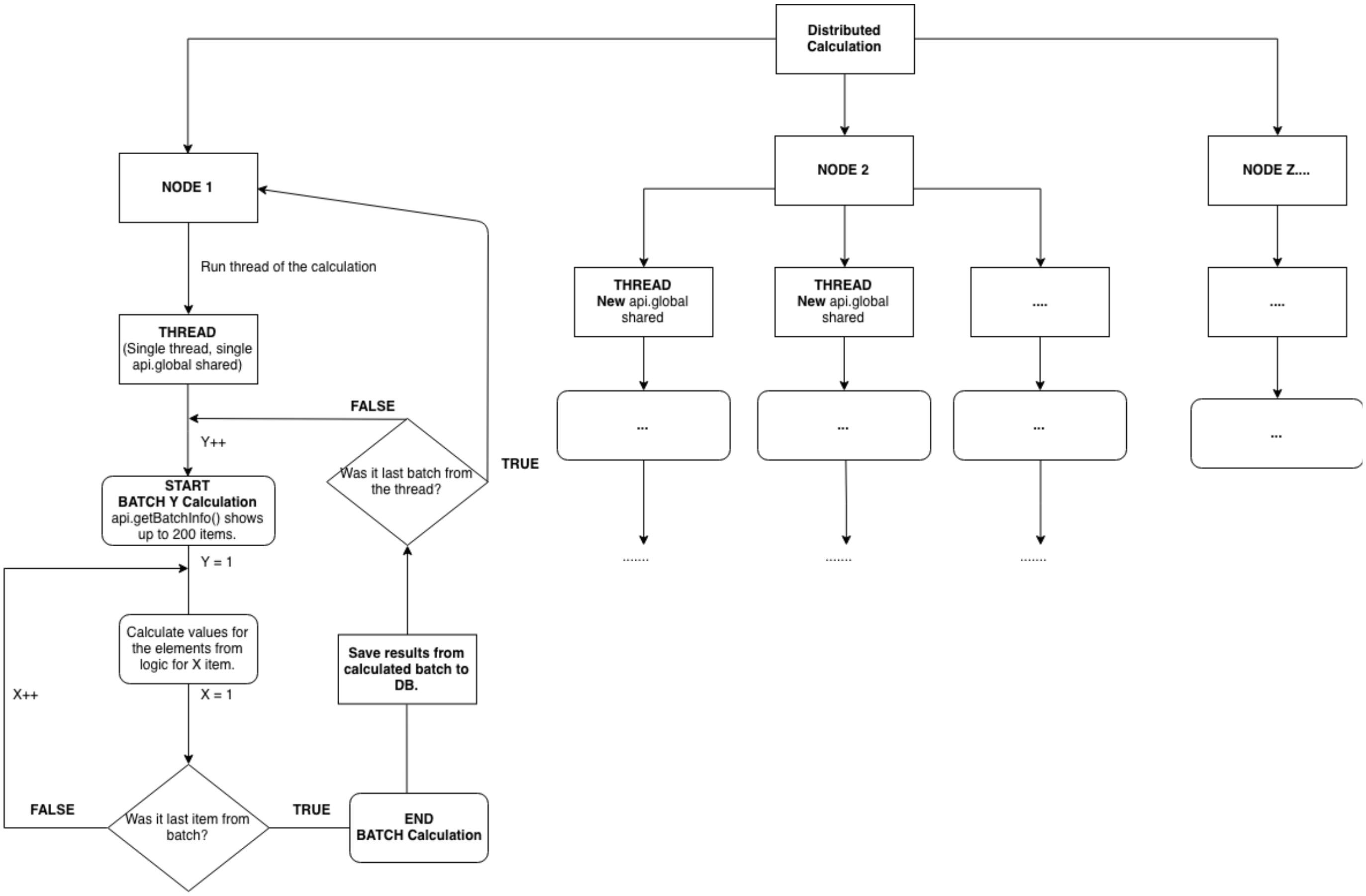
Distributed Calculation

This flow show example of calculation process for distributed calculation.

It was built based on the non distributed calculation flow chart, but the idea is to show the processes only for the one NODE (which in fact might have a lot of more THREADS), but the process is the same on all NODES, the number of NODES for calculation is dynamically created, so it is hard to predict how many NODES you are going to have in the game and how many THREADS in total.

Example:

☑ Allow distributed calculation


Distributed Calculation
NODE 1
NODE 2
NODE Z....
Run thread of the calculation
THREAD (Single thread, single api.global shared)
THREAD New api.global shared
THREAD New api.global shared
....
....
...
...
...
...
FALSE
Y++
Was it last batch from the thread?
TRUE
START BATCH Y Calculation api.getBatchInfo() shows up to 200 items.
Y = 1
....
....
....
Calculate values for the elements from logic for X item.
Save results from calculated batch to DB.
X++
X = 1
FALSE
Was it last item from batch?
TRUE
END BATCH Calculation

Based on this flow, imagine that you will have 6000 items to be calculated, those are going to be split for all of the nodes.

For example if you have 3 nodes in the game, let's say that that this number will be divided by 3, having said NODE1 will have 2000, NODE2 will have 2000, NODE3 will have 2000.

Next step is to split those items for the batches and jobs to be calculated in (each node and thread will work in parallel), having said that there is going to be 5 threads per node, we will end up having:

On NODE1 400 items to be calculated (2 batches) per THREAD, and same for the other ones, meaning that the **api.global will be shared only between THREAD (not NODE)** meaning that you cannot pass items between nodes using api.global, in that case api.getSharedCache()/set.../remove... methods should be used, which then can be accessed between nodes and it's threads.

In the pricefx-config.xml file, you can restrict the number of parallel slave threads for a given job by its size. The definition MUST be sorted from low to high. The key defines the maximum job size for this slave count (="up to" semantics).

```
<maxSlaveThreadsPerJobMap>
  <upto_5000>1</upto_5000>
  <upto_20000>2</upto_20000>
  <upto_50000>3</upto_50000>
  <upto_10000000>4</upto_10000000>
</maxSlaveThreadsPerJobMap>
```

Q&A

- Is there any min limit that will consider if the job will create batches or not?
    - Yes there is, up to the environment it might be vary, please ask support for your specific example.
    - For shared production it was tested well that if there is more than 2000 items, then calculation will create batches and then calculate them, if the limit is not met, batches will always return null and calculation will be only performed in the row calculation level.
- Is it worth using it in all cases?
    - I would say yes, if you are worried about performance, this is the first point that you should refer to.
- I am unable to use api.addOrUpdate() method in distributed calculation, is there any workaround for this function?
    - Yes there is a api.boundCall() which can be called including batches, but it is not recommended, however if you do it safety then it will make sense (addOrUpdate only specific sort of keys, not the same).
    - Please refer to this **link** as example of single usage, or this **link** as example of batching usage.

Coding Style

- Put "def" Outside of Loops
- Many OR Constants

Put "def" Outside of Loops

Since the calculation counts the objects count whenever a new instance is created, you may reach the limit and get an error about too many instances. To avoid this problem, put "def" outside the loops or closures.

So instead of doing:

Too many instances

```
for (record in records) {
  def cost = ....
}
```

Use this approach:

Safer approach

```
def cost
for (record in records) {
  cost = ....
}
```

Many OR Constants

Instead of:

Bad pattern

```
if (color == "Red || color == "Green" || color == "Blue")
```

use:

Bad pattern

```
if (["Red", "Green", "Blue"].contains(color))
```

Naming Conventions

- General
- Logic Names
- Element Names
- Element Labels
- Groovy Variables
- Groovy Methods
- DF/DS/DM Fields

General

Whenever an identifier is required (uniqueName, name, etc.), it is recommended to use identifiers in a ClassicalCamelCaseFormAsYouKnowItFromJava, e.g. SalesPrice. If there is a abbreviation used in the name, it is naturally useful to separate it with an underscore character "_", for example, ABC_Class (instead of ABCClass).

Labels are typically a copy of the uniqueName/name and the words are split using spaces, e. g. Sales Price.

Here are some commonly used conventions for the identifier suffixes:

Suffix	Example	Label	Meaning
Diff	VolumeDiff	Volume	Represents the difference between an actual and previous value
Abs	MarginAbs	Margin	Margin represented as an amount (typically formatted as Money)
Pct	MarginPct	Margin %	Margin represented as a percentage (typically formatted as Percent)
s	pxRecords		Represents a list of values

The suffixes can be combined.

Logic Names

The general rule is to make the first letter uppercase.

Example: DefaultLogic, Pricelist, Main, DefaultQuoteLogic

For the following logic types, there is a recommended prefix/suffix:

Logic type	Formula Nature	Prefix	Suffix	Example
Calculation Flow	`calculationFlow`	CF_		CF_DailyCalculation
Calculated Field Set	(default)	CFS_		CFS_ProductEnrichment
Dataload	`paDataLoad`	DL_		DL_LoadFromPX
Groovy Library	`library`		Lib	SharedLib

Element Names

The general rule is to make the first letter uppercase. The reason is that the element is implemented as a Groovy class and therefore referring to the element's methods from Groovy is done as SomeElement.callSomeMethod(). For this syntax the auto-completion will work in Pricefx Studio, since IDEA will understand it as a static call of a method (even though it is not a static call but an instance call on a binded object). Another reason for the first capital letter is to have the possibility to refer to previous elements: **PM-284** - Getting issue details... STATUS

In order to make the auto-completion work in Studio or to be able to run TDD4C without marking the folders as a source in Studio, it is recommended to always use a different element name which is unique within the project. Also the element name "Filter" is not recommended, since the auto-completion collides with the standard Filter object used frequently by the logics.

The library element names should have a suffix "Utils" (e.g. RoundingUtils, CacheUtils, DatamartUtils).

Example: NewMargin, SalesPrice

Element Labels

Element labels should ideally match the element name where just spaces are added and each word starts with a capital letter (commonly used pattern in UK or US).

The element label should not be identical to a different element name or field name, since functions accept both label and element.

The basic rule is that labels can be adjusted at any point in time and so you should not refer to labels in the logic, instead use the element name or attribute.

Groovy Variables

The general rule is to make the first letter lowercase. Even though Groovy, compared to Java, allows the first letter in uppercase, it is recommended to follow the Java convention.

Example: newMargin, salesPrice

If the variable stores a list of values, then the suffix "s" (English plural) can give a hint that it stores multiple values.

```
def records = api.find(...)

    for (record in records) {
  //do something with with record.attribute1
  ...
 }
```

Groovy Methods

The general rule is to make the first letter lowercase. Even though Groovy, compared to Java, allows the first letter in uppercase, it is recommended to follow the Java convention.

Example: getCostPrice()

DF/DS/DM Fields

The general rule is to make the first letter uppercase.

Note: In SQL queries the fields are referred to in lowercase.

Example: NewMargin, SalesPrice

Backend Error Messages

Error message	Reason
java.util.concurrent.ExecutionException: net.pricefx.service.DMPersistenceServiceException: java.lang.IllegalArgumentException: Schema for partition 'xxxx' does not exist	PriceAnalyzer is not set up.
java.util.concurrent.ExecutionException: net.pricefx.service.ObjectLockedException: Object xxx is locked	
net.pricefx.service.impl.dmsession.FBCException: java.lang.IllegalStateException: Reading from xxx failed; file length -1 read length 12288 at 156382424 [1.4.197/1]	
java.lang.IllegalStateException: This store is closed	There is a bug in the (H2 DB) MVStore lib that we used up to Salty Dog 3.5.5, see: **PFCD-5075** - PA query cache exception: store is closed **QA VERIFIED**

o.h.e.j.spi.SqlExceptionHelper - (conn=360710) Could not send query: query size is >= to max_allowed_packet (16777216)	SQL query is too big.
Thread died	The server was restarted or crashed.
java.util.concurrent.ExecutionException: net.pricefx.service.DMPersistenceServiceException: Calling method 'createQuery' is not valid without an active transaction (Current status: MARKED_ROLLBACK)	This will typically happen if you get a DatamartContext and try to use it later in a differe not to store it or pass it around. If you do not do that, then there was some error that ca rolled back.
Too many new instances created: xxxx	Too many objects were created in the server memory. Sometimes the Groovy engine c instances of the objects.
Maximum In-Memory row count of 100000 exceeded.	The data load job tries to work with more rows than the allowed maximum (100.000).

No content to map due to end-of-input at [Source: okhttp3.ResponseBody$BomAwareReader@50fdc87a; line: 1, column: 0]	The logic which you try to test execute has a bug.
	From the MS project:
	`api.trace("THIS", "FrozenLastFinalPrice", FrozenLastFinalPrice)`
	`BigDecimal FrozenLastFinalPrice = api.getElement("FrozenLastFinalPrice")?.toB`
	FrozenLastFinalPrice was not declared before the API trace call. In the Salty Dog versi deploying the next version El Presidente we get this issue. It was resolved by fixing the

Customer Volume Metrics

Project	Rows in Transaction DM	Columns in Transaction DM	Rows in PM	Attributes in PM	Rows in CM	Attributes in CM	PXs	Max Rows in PXs	Attributes In PX with Max Rows	CXs	Max Rows in CXs	Attributes In CX with Max Rows	PPs	Max Rows in PPs	Attributes In PP with Max Rows	Quotes	Max Line Items in Quote	Quote Calculation Results	RBAs
AD-NA	1,466,589	128	26,208	30	766	25	5	22,568	1	8	618,025	47	80	20,276	28	1,001		59	69
Solenis	N/A	N/A	69,900	18	144,319	29	19	251,572	12	3	1,526,832	19	33	34,207	4	31,039		43	N/A