

MRMCashFlowComponent

August 17, 2026

MRMCashFlowComponent

Kind: Class

Source: Pams/Web/Pams.Web/Client/app/components/dashboard/dashboard-library/widgets/mrm-cash-flow/mrm-cash-flow.component.ts (line 145)

Part of: [Pams](#)

`MRMCashFlowComponent` manages the cash-flow widget in the dashboard. It loads and initializes dashboard data, builds monthly summaries, calculates totals by month and year, saves changes, and reads the currency rate from dashboard header data.

Methods

Method	Signature	Returns
<code>ngOnInit</code>	<code>ngOnInit()</code>	<code>void</code>
<code>ngOnChanges</code>	<code>ngOnChanges()</code>	<code>void</code>
<code>ngOnDestroy</code>	<code>ngOnDestroy()</code>	<code>void</code>
<code>getDashboardData</code>	<code>getDashboardData()</code>	<code>void</code>
<code>fillMonthsSummery</code>	<code>fillMonthsSummery()</code>	<code>void</code>
<code>initalizingData</code>	<code>initalizingData(value: BranchData)</code>	<code>void</code>
<code>saveDashboardData</code>	<code>saveDashboardData()</code>	<code>void</code>
<code>calculateSumByYear</code>	<code>calculateSumByYear()</code>	<code>void</code>
<code>calculateSumByMonth</code>	<code>calculateSumByMonth(monthIndex: number)</code>	<code>void</code>
<code>getCurrencyRateFromHeader</code>	<code>getCurrencyRateFromHeader(currencyCode: string)</code>	<code>number</code>
<code>setCurrencyTotalToHeader</code>	<code>setCurrencyTotalToHeader(cashFlowObjIndex: number, currencyCode: string, amount: number)</code>	<code>void</code>

Method	Signature	Returns
resetCurrencyTotal	resetCurrencyTotal(cashFlowObjIndex: number)	void
calculateEditingSumByMonth	calculateEditingSumByMonth(cashFlowObjIndex: number, itemIndex: number, currencyIndex: number, monthIndex: number)	void
translateMonthName	translateMonthName(MonthId: number)	string
getpriceFormatted	getpriceFormatted(x: undefined)	string
onResize	onResize()	void
setChaching	setChaching()	void
getChaching	getChaching()	void
switchVendorTypeIdHandler	switchVendorTypeIdHandler()	void
switchByGroupsHandler	switchByGroupsHandler()	void
switchBasedOnTendencyHandler	switchBasedOnTendencyHandler()	void
comparingWithPrevMonth	comparingWithPrevMonth(month: CashFlowItemMonthlyDetail, cashFlowObjIndex: number, monthIndex: number, propertyName: string)	string
comparingSummeryWithPrevMonth	comparingSummeryWithPrevMonth(month: MonthsSum, monthIndex: number, propertyName: string)	string
comparingEditingWithPrevMonth	comparingEditingWithPrevMonth(month: CashFlowItemMonthlyDetail, cashFlowObjIndex: number, itemIndex: number, currencyIndex: number, monthIndex: number, propertyName: string)	string
isCurrentMonth	isCurrentMonth(month: CashFlowItemMonthlyDetail)	boolean
expectedCurrencyChange	expectedCurrencyChange(e: undefined)	void
selectText	selectText(e: undefined)	void
editSelectedMonthPrepare	editSelectedMonthPrepare(month: MonthsSum, supplierId: number)	void
fromNow	fromNow(DateTime: any)	void

Method	Signature	Returns
mouseoverRowSpan	mouseoverRowSpan(e: undefined, bIndex: number)	void
mouseoutRowSpan	mouseoutRowSpan(e: undefined, bIndex: number)	void
openPopup	openPopup(n: string, t: string, i: string, btnTxt: string)	void
savePopup	savePopup()	void
ClosewithDonotSave	ClosewithDonotSave()	void
OnOpenedPopup	OnOpenedPopup()	void
closePopup	closePopup()	void

Properties

Property	Type
sharedPopup	SharedPopupComponent
_widget	Widget
_widgetType	string
withoutOptionBar	boolean
editMode	boolean
cashFlowDisplay	number
bookingOrSales	number
byGroups	boolean
isPamsData	boolean
selectedTeam	number
forecastRuleId	number
data	BranchData
allowedtoEdit	boolean
allowedtoEnableEdting	boolean
widgetEditRequest	any
widgetEditEnd	any

Property	Type
forecastRules	any[]
currencies	CompanyCurrency[]
selectedYear	number
currentMonthIndex	number
numYearsDatasource	any[]
todayWidgetsEnum	any
cachingWidgetData	CachingWidgetData
originalData	BranchData
selectedData	BranchData
monthsSummery	MonthsSum[]
branchCurrency	any
salesCurrency	any
basedOnTendency	boolean
subscription	Subscription
SeriseRange	any[]
echartsIntance	any
totalTargets	number
yearBooking	number
totalForecasts	number
tempTotalTarget	TotalTargets
totalProfitAndLossAmount	number
totalOthers	number
totalTotalSGAndA	number
totalFORX	number
totalTotal	number
totalCumulative	number
totalPercentageUsedBudget	number
popup	any

Property	Type
searchText	string

Where it refuses work

- `MRMCashFlowComponent` stops the work with an early return when `findMonthIndex > -1` .
- `MRMCashFlowComponent` stops the work with an early return when `MonthId == 12` .
- `MRMCashFlowComponent` stops the work with an early return when `month[propertyName] < this.data.BanksSummary[monthIndex - 1][propertyName]` .
- `MRMCashFlowComponent` stops the work with an early return when `month[propertyName] < this.data.ReceivablesSummary[monthIndex - 1][propertyName]` .
- `MRMCashFlowComponent` stops the work with an early return when `month[propertyName] < this.data.PayablesSummary[monthIndex - 1][propertyName]` .
- `MRMCashFlowComponent` stops the work with an early return when `month[propertyName] < this.data.LGsSummary[monthIndex - 1][propertyName]` .

Diagram

```
graph LR
  Init[ngOnInit / ngOnChanges] --> Load[getDashboardData]
  Load --> Initialize[initializingData]
  Initialize --> Summary[fillMonthsSummary]
  Summary --> MonthTotals[calculateSumByMonth]
  Summary --> YearTotals[calculateSumByYear]
  Initialize --> Currency[getCurrencyRateFromHeader]
  Save[saveDashboardData] --> Load
  Destroy[ngOnDestroy] --> Cleanup[Release subscriptions or state]
```

Usage

```
import { TestBed } from '@angular/core/testing';
import { MRMCashFlowComponent } from './mrm-cash-flow.component';

const fixture = TestBed.createComponent(MRMCashFlowComponent);
const component = fixture.componentInstance;

// Angular runs lifecycle hooks during change detection.
fixture.detectChanges();

// Refresh dashboard cash-flow data after an external dashboard update.
component.getDashboardData();
```

```
// Persist changes made to the widget data.  
component.saveDashboardData();
```

AI Coding Instructions

- Keep data loading in `getDashboardData()` and initialize derived widget state through `initializingData()` .
- Call `fillMonthsSummery()` after changing source cash-flow data so month totals remain aligned with the displayed data.
- Use `calculateSumByMonth()` and `calculateSumByYear()` for totals rather than duplicating aggregation logic in templates.
- Read exchange-rate values through `getCurrencyRateFromHeader()` so calculations use dashboard header configuration.
- Preserve cleanup behavior in `ngOnDestroy()` when adding subscriptions, timers, or other component resources.

Used by

2 references from 2 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (2)

- `TranslateHandler` — `Pams/Web/Pams.Web/Client/app/app.module.ts :1`
- `ReportsTabComponent` — `Pams/Web/Pams.Web/Client/app/components/dashboard/reports-library/reports-tab/reports-tab.component.ts :1`