

XIRR (not yet available)

🌐 Language: [Deutsch](#) · [English](#)

💡 XIRR (not yet available)

Data sheet and documentation entry point for the XIRR Nuclet (calculation of the internal rate of return (XIRR)).

CONCEPT

INTEGRATORS

UPDATED: JUL 2026

APPLIES TO NUCLOS 4.2026.X

On this page

- [Overview](#)
- [Integration](#)
- [Version history](#)
- [Related pages](#)



Machine-translated – under review

This page was machine-translated from German as a first draft and is currently under review. The authoritative source remains the **German original** (see the language switch above).

- [XIRR \(not yet available\)](#)
 - [Overview](#)
 - [Short description](#)
 - [Nuclet components](#)
 - [Java package structure](#)
 - [Integration](#)
 - [Necessary integration steps](#)
 - [Version history](#)
 - [Related pages](#)
 - [Supporting Nuclets](#)
 - [Nuclet Wiki](#)

Name:	XIRR
Package:	org.nuclet.xirr
Version:	1.4.0
Date:	27.09.2017
Nuclos compatibility:	from Nuclos 4.19.3

Overview

Short description

The XIRR Nuclet offers functions for calculating the internal rate of return. The calculation is based on a series of payments (amounts of money, bound to date values).

The calculation corresponds to the Excel functions XIRR (or XINTZINSFUSS).



The calculation is based on an initial start or estimated value. For a successful calculation, this must lie within a convergence range.

Nuclet components

Within the .nuclet file, the XIRR Nuclet comprises

- various Java rules (distributed across packages),
- a job control (for controlling a test job).

Type	Name, English	Name, German	Short description
Java package	in.satpathy.financial		XIRR functionality used internally by the Nuclet
	in.satpathy.math		XIRR functionality used internally by the Nuclet
	org.nuclet.xirr.data		XIRR data structures
	org.nuclet.xirr.logic		rule interface for XIRR functions
	org.nuclet.xirr.parameter		system parameter definitions
	org.nuclet.xirr.test		test routines
Job control	XIRR Test		test job for calling test routines

Table 1: Nuclet components

Java package structure

The Java rules are divided into five packages. Of these rules, normally only four Java classes are really relevant for the Nuclet integration:

- org.nuclet.xirr.data.XIRRSeries, as a data structure for a series of payments
- org.nuclet.xirr.logic.XIRRLogic, as logic for calling the XIRR calculations
- org.nuclet.xirr.test.XIRRTestJob, as a job for calling tests
- org.nuclet.xirr.parameter.XIRRSystemParameters, for configuring system parameters

Integration

Necessary integration steps

The integration of the XIRR Nuclet takes place in 3 steps:

1. Download
2. Nuclet import
3. Configuration of system parameters
4. Connecting your own Java code to the XIRR logic

All integration steps are explained in detail below.

Step 1: Download

Download the ZIP file „XIRR-v1.4.0-4.19.3.zip“ from the Nuclos web page under „Nuclos Services“ > „Download“ > „Nuclet Download“. Then unpack the ZIP locally.

Step 2: Nuclet import

Import the XIRR Nuclet under „Configuration“ > „Nuclet Management“ > „Import“ into your existing Nuclos instance, selecting the file „XIRR-v1.4.0-4.19.3.nuclet“

Step 3: Configuration of the system parameters

In version v1.0.0, the used Nuclos system parameter is set up by adjusting the constant XIRR_DEFAULT_GUESS in the class org.nuclet.xirr.parameter.XIRRSystemParameters:

Parameter	Short description
XIRR_DEFAULT_GUESS	Default start value for the XIRR iterations

Table 2: System parameters



Note: In a future version – i.e. as soon as the new Nuclos API supports reading system parameters – this step will no longer run via an adjustment of the source code.

Step 4: Connecting your own Java code to the XIRR logic

The XIRR logic is integrated by initialising an org.nuclet.xirr.data.XIRRSeries, which is filled with the payment amounts and calendar dates of the payments. Optionally, a specific start value („guess“) can be passed. If no start value is specified, the default start value is used (see step 3). The series object is then passed to an instance of the class org.nuclet.xirr.logic.XIRRLogic for the calculation.

Here is an example of integration in an UpdateRule:

```

@Rule(name="AktualisiereZahlungsverkehr", description="Aktualisiert den Zahlungsverkehr")
public class AktualisiereZahlungsverkehr implements UpdateRule
{
    public void update(UpdateContext context)
    {
        (...)

        XIRRLogic logic = new XIRRLogic(context); // Initialisierung der XIRR-Logik
        XIRRSeries series = new XIRRSeries(); // es ist möglich, den Startwert (guess) mizugegeben

        Zahlungsverkehr zvk = context.getBusinessObject(Zahlungsverkehr.class);
        List<Zahlung> lstZahlungen = zvk.getZahlung();

        for (Zahlung z : lstZahlungen) {
            series.add(z.getDatum(), z.getBetrag());
        }

        BigDecimal bgdXirrValue = logic.calculate(series); // ohne Konvergenz ist der Wert „null“!

        (...)
    }
}

```

 **Note:** See also the method test2() in the class org.nuclet.xirr.test.XIRRTestJob.

Version history

Version	Date	Type	Changes
1.0.0	19.04.2013	initial version	-
1.1.0	13.12.2013	Migration to Nuclos 4.0	• Migration to Nuclos 4.0 • in particular: migration to Nuclet parameters
1.2.0	24.03.2017	Migration to Nuclos 4.13	Migration to Nuclos 4.13
1.3.0	22.08.2017	Migration to Nuclos 4.18	Migration to Nuclos 4.18
1.4.0	27.09.2017	Migration to Nuclos 4.19	Migration to Nuclos 4.19

Related pages

Supporting Nuclets

Back to Supporting Nuclets

[Open](#)

Nuclet Wiki

Nuclet Wiki home

[Open](#)