

1 MT940: Überblick

 Sprache: **Deutsch** · [English](#)

1 MT940: Überblick

Das MT940-Nuclet liest elektronische Kontoauszüge im MT940-Format automatisiert in Nuclos ein.

KONZEPT

INTEGRATOREN

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

Auf dieser Seite

- [Verwandte Seiten](#)

Kurzbeschreibung

Das MT940-Nuclet bietet die Möglichkeit

- elektronische Kontoauszüge, die im MT940-Format vorliegen, automatisiert einzulesen
- die eingelesenen Daten in eine Struktur von Nuclos-Businessobjekten zu übertragen
- die Bankumsätze anhand des Verwendungszwecks automatisch andere Businessobjekte ("Referenzen", das sind je nach Anwendungszweck bspw. Ihren Rechnungen) zuzuordnen

Für die den Bankumsätzen zugeordneten Objekten (Rechnungen, o.ä.) wird bei Zahlungseingang automatisiert ein Statuswechsel durchgeführt (bspw. von "offen" nach "bezahlt").

Mehr zu Funktionsumfang und implementierten Abläufen dieses Nuclets finden Sie im [Abschnitt 2](#).

Nuclet-Bestandteile

Das MT940-Nuclet umfasst im Rahmen der .nuclet-Datei

- sieben Businessobjekte (für Kontoauszüge, Bankumsätze und diverse Stammdaten),
- sechs Layouts,
- ein Statusmodell (für Bankumsätze),
- zwei Attributgruppen,
- diverse Java-Regeln (verteilt auf Packages),
- drei Nuclet-Parameter,
- eine Jobsteuerung (zur Steuerung des Import-Jobs),
- drei Strukturdefinitionen (für den Import von Stammdaten) und
- eine Nuclet-Abhängigkeit.

Darüberhinaus werden in der ZIP-Datei folgende Komponenten mitgeliefert:

- drei CSV-Dateien für Objektimporte sowie
- zwei MT940-Beispieldateien mit Kontoauszügen.

Typ	Name, englisch	Name, deutsch	Kurzbeschreibung
Businessobjekt	Bank Statement	Kontoauszug	
	Bank Transaction	Bankumsatz	
	Bank Transaction Type	Bankgeschäftsvorfall	
	Banking Business Line	Bankgeschäftssparte	
	Debit/Credit Mark	Soll-Haben-Kennung	
	MT940 Reference	MT940-Referenz	
	Bank Transaction Ref	Referenz-2-Bankumsatz	
Layout	Bank Statement		Layout für Businessobjekt „Bank Statement“
	Bank Transaction		Layout für das Businessobjekt „Bank Transaction“
	Bank Transaction Type		Layout für Businessobjekt „Bank Transaction Type“
	Banking Business Line		Layout für Businessobjekt „Banking Business Line“

	Debit/Credit Mark		Layout für Businessobjekt „Debit/Credit Mark“
Statusmodell	MT940 Bank Transaction		Statusmodell für das Businessobjekt „Bank Transaction“
Attributgruppe	Bank Transaction Data		Bankumsatz- und Kontoauszugsdaten
	Bank Transaction References		den Bankumsätzen zugeordnete Referenzen
Java-Package	org.nuclet.mt940.facade		Klassen für Datenbankzugriffe
	org.nuclet.mt940.job		Java-Regeln zur Steuerung von Jobs („Jobsteuerung“)
	org.nuclet.mt940.logic		Geschäftslogik
	org.nuclet.mt940.parser		MT940-Parserfunktionalität
	org.nuclet.mt940.rule		Steuerung von Insert-/Update-/Delete-Events
	org.nuclet.mt940.statemodel		Definition von Statuswechseln (als Nuclet-Schnittstelle)
	org.nuclet.mt940.wrapper		Wrapper-Klassen (als Nuclet-Schnittstelle)
Parameter	MT940 File Directory		legt das Eingangsverzeichnis der MT940-Dateien fest
	MT940 File Encoding		bestimmt den Zeichensatz, der in den MT940-Dateien genutzt wird
	MT940 File Extension		definiert die Dateinamenerweiterung der MT940-Dateien
Jobsteuerung	MT940 Import		Fristenjob für den automatisierten MT940-Import
Strukturdefinition	MT940 Bank Transaction Type		Importstruktur für Businessobjekt „Bank Transaction Type“
	MT940 Banking Business Line		Importstruktur für Businessobjekt „Banking Business Line“
	MT940 Debit/Credit Mark		Importstruktur für Businessobjekt „Debit/Credit Mark“
Nuclet-Abhängigkeit	org.nuclet.Common		allgemeine Helferfunktionalität
Objektimporte	Banking_Business_Line.csv		Stammdatensätze für Businessobjekt „Banking Business Line“
	Bank_Transaction_Type.csv		Stammdatensätze für Businessobjekt „Bank Transaction Type“
	Debit_Credit_Mark.csv		Stammdatensätze für Businessobjekt „Debit/Credit Mark“
Beispieldateien	Kontoauszug_001.sta		MT940-Importbeispiel
	Kontoauszug_002.sta		MT940-Importbeispiel

Tabelle 1: Nuclet-Bestandteile



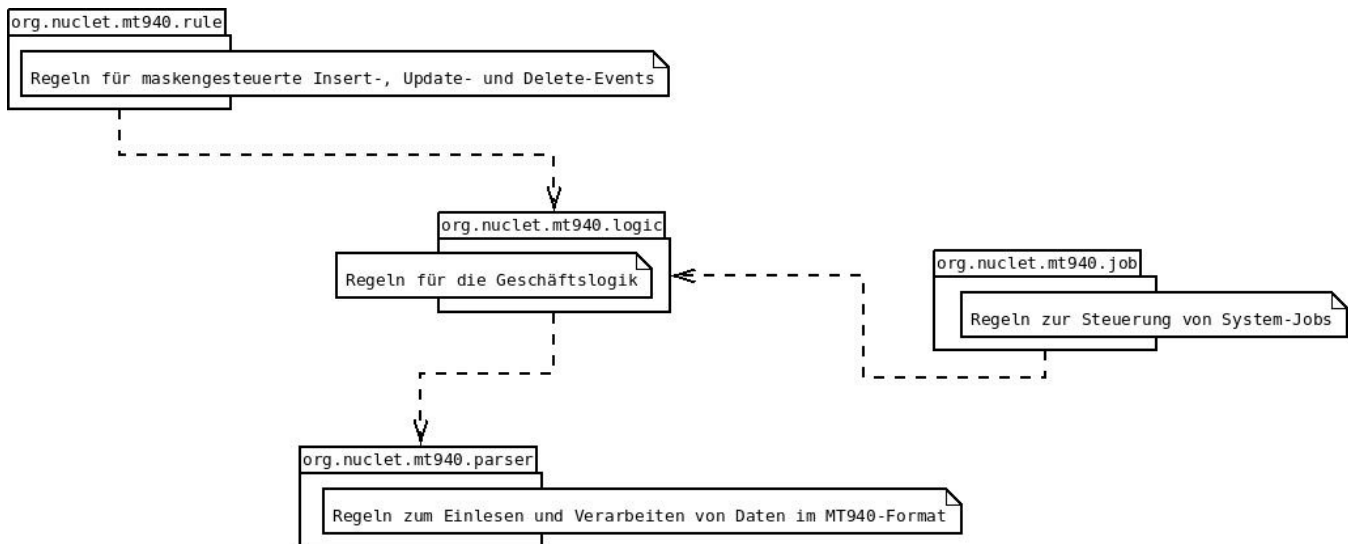
Achtung: Bitte beachten Sie, dass durch die Nuclet-Abhängigkeit zu [org.nuclet.Common](#) bei einer Integration des Nuclets weitere Komponenten ins System importiert werden, die hier nicht im einzelnen aufgeführt werden. Details zu den Bestandteilen des [Common-Nuclets](#) finden Sie in der Dokumentation dieses Nuclets.

Java-Package-Struktur

Die Java-Regeln sind in sieben Packages unterteilt:

- Regeln für maskengesteuerte Events ([org.nuclet.mt940.rule](#))
- Regeln zur Steuerung von System-Jobs ([org.nuclet.mt940.job](#))
- Regeln für die Geschäftslogik ([org.nuclet.mt940.logic](#))
- Regeln zum Einlesen und Verarbeiten von MT940-Dateien ([org.nuclet.mt940.parser](#))
- Regeln für Datenbankzugriffe ([org.nuclet.mt940.facade](#))
- Wrapper-Klassen als Nuclet-Schnittstelle ([org.nuclet.mt940.wrapper](#))
- Statuswechsel-Definitionen als Nuclet-Schnittstelle ([org.nuclet.mt940.statemodel](#))

Die wesentlichen Abhängigkeiten der Packages sind in der folgenden Abbildung veranschaulicht. Die Abbildung beschränkt sich auf diejenigen Hauptpakete, die der Prozesssteuerung dienen.



In drei dieser vier Java-Packages stellt eine abstrakte Java-Klasse die Basisfunktionalität über die deklarierten und (größtenteils) definierten Methoden. Die Import-Prozesse sind entlang der vordeklarierten Methoden dieser abstrakten Klassen festgelegt und bedürfen für den normalen Anwendungsfall keiner weiteren Anpassung.

In den Paketen für Datenbankzugriffe (org.nuclet.mt940.facade) und Wrapper-Klassen (org.nuclet.mt940.wrapper) sind ebenfalls abstrakte Klassen definiert, deren konkrete Ausprägungen anwendungsspezifisch zu implementieren sind. Konkret bedeutet dies an dieser Stelle, Schnittstellen zum Zielnuclet herzustellen und anwendungsspezifisches Verhalten festzulegen (und zwar für Währung, Zahlungsbedingungen und Referenzen).

Java-Package	abstrakte Klassen	konkrete Implementierungen
org.nuclet.mt940.job	AbstractMT940Importer	MT940Importer
org.nuclet.mt940.logic	AbstractMT940Logic	MT940Logic
org.nuclet.mt940.parser	AbstractMT940Parser	- MT940Parser - MT940SparkasseParser
org.nuclet.mt940.rule	-	-
org.nuclet.mt940.facade	- AbstractCurrencyFacade - AbstractReferenceFacade	- CurrencyFacade - ReferenceFacade
org.nuclet.mt940.wrapper	- AbstractConditionsOfPaymentWrapper - AbstractCurrencyWrapper - AbstractReferenceWrapper	- ConditionsOfPaymentWrapper - CurrencyWrapper - ReferenceWrapper
org.nuclet.mt940.statemodel	-	

Tabelle 2: Java-Package-Struktur

Anwendungsspezifische Anpassungen sollten nur umgesetzt werden, entweder

- in den konkreten, bereits vorhandenen, Klassen (siehe Tabelle) oder
- durch neue, eigene konkrete Implementierungen

Das Package org.nuclet.mt940.parser enthält die Klasse MT940SparkasseParser, die bereits dezidiertes Verhalten für von Sparkassen gelieferte MT940-Dateien zur Verfügung stellt. Die zweite konkrete Implementierung innerhalb dieses Packages ist die Klasse MT940Parser, die über die Basisfunktionalität des AbstractMT940Parser hinaus kein zusätzliches Verhalten mitbringt. Für MT940-Dateien, die von anderen Kreditinstituten erstellt werden, müsste die Klasse MT940Parser (oder eine eigene, konkrete Implementierung des AbstractMT940Parser) ggf. mit bankspezifischem Verhalten gefüllt werden.

Das MT940-Nuclet ist zur Zeit derart konfiguriert, dass das sparkassenspezifische Verhalten genutzt wird. Die Initialisierung des Parsers erfolgt im Konstrukt der Klasse org.nuclet.mt940.job.MT940Importer und könnte dort jederzeit auf ein anderes Verhalten, d.h. eine eigene Implementierung umdirigiert werden.

org.nuclet.mt940.job.MT940Importer

```
public MT940Importer(final JobContext context,
    final String strMT940Directory,
    final String strMT940ReferenceType)
{
    super(context, strMT940Directory, strMT940ReferenceType);

    // @replace!
    //
    // Replace with your own logic and/or your own parser here, if you
    // need more specific behaviour.
    //
    this.logic = new MT940Logic(context, strMT940ReferenceType);

    this.parser = new MT940SparkasseParser(context);
}
```

Der Java-Sourcecode ist mit @replace!-Tags an all jenen Stellen markiert, wo anwendungsspezifisches Verhalten hinzugefügt werden kann. Dazu mehr im [Abschnitt 4](#).

Verwandte Seiten



Nuclet: MT940(noch nicht verfügbar)

MT940-Datenblatt

[Öffnen](#)



Schnittstellen

Alle Integrationen

[Öffnen](#)