

4.7.2 MT940: Business object facades

Language: Deutsch · English

4.7.2 MT940: Business object facades

Business object facades – part of the MT940 Nuclet documentation (import and processing of bank statements in MT940 format).

REFERENCE INTEGRATORS UPDATED: JUL 2026 APPLIES TO NUCLOS 4.2026.X

On this page

- 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).

The MT940 Nuclet works with wrapper objects instead of the BusinessObjects known only to the target Nuclet (see section 4.7.1). Therefore, when integrating the MT940 Nuclet, it must be ensured that certain database accesses can operate on the actually used business objects. This means the classes and attribute names used must be known. This is provided via business-object-specific facades.

In addition, the facades provide wrapper methods that convert the actually used business objects into the wrapper objects defined in section 4.7.1.

When connecting the MT940 Nuclet, the following facade classes for currency objects and references must be adjusted.

Facade class	Function	Java package	Methods to adjust
CurrencyFacade	definition of classes and attributes of the currency object	org.nuclet.mt940.facade	- getWrappedClass() - getIso4217CodeAttribute()
ConditionsOfPaymentFacade	definition of the conditions of payment class	org.nuclet.mt940.facade	getWrappedClass()
ReferenceFacade	- definition of classes and attributes of the reference object - definition of state changes	org.nuclet.mt940.facade	- getWrappedClass() - getNuclosStateAttribute() - getSourceStates() - getDestinationState()

Table 4.7.2: Overview, adjustments in facade classes

4.7.2.1 CurrencyFacade

The methods getWrappedClass() and getIso4217CodeAttribute() must be implemented in the CurrencyFacade class:

- getWrappedClass() should return the Java class of the currency object used (e.g. org.nuclet.currency.Currency)
- getIso4217CodeAttribute() should return the attribute of the currency object used that represents the ISO 4217 code (e.g. prg.nuclet.currency.Currency.Iso4217Code)

An example is given in each case in a comment block; this example must be adapted to the actually used currency business object (or its BusinessObject class).



Please also note that the generic "<T>" must be replaced accordingly in lines 42 and 76.

org.nuclet.mt940.facade.CurrencyFacade

```
package org.nuclet.mt940.facade;
```

```

import java.util.List;

import org.nuclos.api.businessobject.BusinessObject;
import org.nuclos.api.businessobject.Query;
import org.nuclos.api.businessobject.QueryOperation;
import org.nuclos.api.businessobject.SearchExpression;
import org.nuclos.api.businessobject.attribute.StringAttribute;
import org.nuclos.api.context.JobContext;
import org.nuclos.api.context.RuleContext;
import org.nuclos.api.exception.BusinessException;
import org.nuclos.api.provider.QueryProvider;
import org.nuclos.api.provider.BusinessObjectProvider;

import org.nuclet.common.facade.AbstractFacade;
import org.nuclet.mt940.facade.AbstractCurrencyFacade;
import org.nuclet.mt940.wrapper.AbstractCurrencyWrapper;
import org.nuclet.mt940.wrapper.CurrencyWrapper;

// @replace! Bitte bei Nuclet-Integration mit eigenem Code ersetzen!
//
// Beispiel:
//
// import org.nuclet.currency.Currency;

/**
 * Facade for Business Objects of type "Currency"
 *
 *
 * @version 1.1
 * @date 20.02.2014
 * @nuclet org.nuclet.MT940
 * @nucletversion 1.4.0
 * @sincenucletversion 1.1.0
 * @since 07.08.2013
 * @author frank.lehmann@nuclos.de
 *
 */
public class CurrencyFacade<T extends BusinessObject> extends AbstractCurrencyFacade<T> // @replace
AbstractCurrencyFacade<Currency>
{
    private static final CurrencyFacade instance = new CurrencyFacade();

    /**
     * Liefert die Singleton-Instanz dieser Klasse
     *
     */
    public static CurrencyFacade getInstance()
    {
        return instance;
    }

    /**
     * Wrap the actual currency objects inside an <code>AbstractCurrencyWrapper</code>
     *
     * @return the actual currency objects, wrapped inside an <code>AbstractCurrencyWrapper</code>
     *
     */
    public AbstractCurrencyWrapper getCurrencyWrapper(final BusinessObject boCurrency)
    {
        // @replace!
        //
        // return new CurrencyWrapper(boCurrency);

        return new CurrencyWrapper(boCurrency);
    }

    /**
     * Get the class of the actual currency objects, i.e. the class wrapped by <code>CurrencyWrapper</code>
     *
     * @return the class of the actual currency objects, i.e. the class wrapped by <code>CurrencyWrapper</code>

```

```

    *
    */
    public Class<T> getWrappedClass() // @replace Class<Currency> getWrappedClass()
    {
        // @replace!
        //
        // return Currency.class;

        return null;
    }

    /**
     * Get the attribute <code>NuclosState</code> of the actual currency objects
     *
     * @return the attribute <code>NuclosState</code> of the actual reference objects
     *
     */
    public StringAttribute getIso4217CodeAttribute()
    {
        // @replace! Bitte bei Nuclet-Integration mit eigenem Code ersetzen!
        //
        // Beispiel:
        //
        // Currency.Iso4217Code;

        return null;
    }
}

```

4.7.2.2 ConditionsOfPaymentFacade

The getWrappedClass() method must be implemented in the ConditionsOfPaymentFacade class:

- getWrappedClass() should return the Java class of the conditions of payment used

An example is given in the comment block; this example must be adapted to the actually used conditions of payment (or its BusinessObject class).



Please also note that the generic "<T>" must be replaced accordingly in lines 34 and 68.

org.nuclet.mt940.facade.ConditionsOfPaymentFacade

```

package org.nuclet.mt940.facade;

import java.util.List;

import org.nuclos.api.businessobject.BusinessObject;
import org.nuclos.api.exception.BusinessException;

import org.nuclet.common.facade.AbstractFacade;
import org.nuclet.mt940.facade.AbstractConditionsOfPaymentFacade;
import org.nuclet.mt940.wrapper.AbstractConditionsOfPaymentWrapper;
import org.nuclet.mt940.wrapper.ConditionsOfPaymentWrapper;

// @replace! Bitte bei Nuclet-Integration mit eigenem Code ersetzen!
//
// Beispiel:
//
// import org.nuclet.businessobject.ConditionsOfPayment;

/**
 * Facade for Business Objects of type "ConditionsOfPayment"
 *
 *
 *
 * @version 1.0
 * @date 07.07.2014

```

```

* @nuclet org.nuclet.MT940
* @nucletversion 1.4.1
* @sincenucletversion 1.4.1
* @since 07.07.2014
* @author frank.lehmann@nuclos.de
*
*/
public class ConditionsOfPaymentFacade<T extends BusinessObject> extends AbstractConditionsOfPaymentFacade<T>
// @replace! AbstractConditionsOfPaymentFacade<ConditionsOfPayment>
{
    private static final ConditionsOfPaymentFacade instance = new ConditionsOfPaymentFacade();

    /**
     * Liefert die Singleton-Instanz dieser Klasse
     *
     */
    public static ConditionsOfPaymentFacade getInstance()
    {
        return instance;
    }

    /**
     * Wrap the actual currency objects inside an <code>AbstractCurrencyWrapper</code>
     *
     * @return the actual currency objects, wrapped inside an <code>AbstractCurrencyWrapper</code>
     *
     */
    public AbstractConditionsOfPaymentWrapper getConditionsOfPaymentWrapper(final BusinessObject
boConditionsOfPayment)
    {
        // @replace!
        //
        // return new ConditionsOfPaymentWrapper(boCurrency);

        return new ConditionsOfPaymentWrapper(boConditionsOfPayment);
    }

    /**
     * Get the class of the actual payment condition objects, i.e. the class wrapped by
<code>ConditionsOfPaymentWrapper</code>
     *
     * @return the class of the actual payment condition objects, i.e. the class wrapped by
<code>ConditionsOfPaymentWrapper</code>
     *
     */
    public Class<T> getWrappedClass() // @replace! public class<ConditionsOfPayment> getWrappedClass()
    {
        // @replace!
        //
        // return ConditionsOfPayment.class;

        return null;
    }
}

```

4.7.2.3 ReferenceFacade

The methods `getWrappedClass()` and `getNuclosStateAttribute()` must be implemented in the `ReferenceFacade` class:

- `getWrappedClass()` should return the Java class of the reference object used (e.g. `org.nuclet.businessobject.ClientBilling`)
- `getNuclosStateAttribute()` should return the attribute of the reference object used that represents the Nuclos status

An example is given in each case in a comment block; this example must be adapted to the actually used currency business object (or its `BusinessObject` class).

The description for adapting the two methods `getSourceStates()` and `getDestinationStates()` [can be found in the next section \(4.7.3\)](#).



Please also note that the generic "<T>" must be replaced accordingly in lines 48 and 139.

org.nuclet.mt940.facade.ReferenceFacade

```
package org.nuclet.mt940.facade;

import java.util.ArrayList;
import java.util.List;

import org.nuclos.api.UID;
import org.nuclos.api.businessobject.BusinessObject;
import org.nuclos.api.businessobject.Query;
import org.nuclos.api.businessobject.QueryOperation;
import org.nuclos.api.businessobject.SearchExpression;
import org.nuclos.api.businessobject.attribute.PrimaryKeyAttribute;
import org.nuclos.api.businessobject.facade.Stateful;
import org.nuclos.api.context.JobContext;
import org.nuclos.api.context.RuleContext;
import org.nuclos.api.exception.BusinessException;
import org.nuclos.api.provider.QueryProvider;
import org.nuclos.api.provider.BusinessObjectProvider;
import org.nuclos.api.statemodel.State;

import org.nuclet.common.facade.AbstractFacade;

import org.nuclet.mt940.facade.AbstractReferenceFacade;
import org.nuclet.mt940.statemodel.ReferenceStatemodel;
import org.nuclet.mt940.wrapper.AbstractReferenceWrapper;
import org.nuclet.mt940.wrapper.ReferenceWrapper;

// @replace! Bitte bei Nuclet-Integration mit eigenem Code ersetzen!
//
// Beispiel:
//
// import org.nuclet.buinessstemplate.ClientBilling;

/**
 * Facade for Business Objects of type "Reference"
 *
 *
 * @version 1.1
 * @date 20.02.2014
 * @nuclet org.nuclet.MT940
 * @nucletversion 1.4.0
 * @sincenucletversion 1.2.0
 * @since 20.09.2013
 * @author frank.lehmann@nuclos.de
 */
public class ReferenceFacade<T extends BusinessObject & Stateful> extends AbstractReferenceFacade<T> //
@replace AbstractReferenceFacade<ClientBilling>
{
    private static final ReferenceFacade instance = new ReferenceFacade();

    /**
     * Liefert die Singleton-Instanz dieser Klasse
     *
     */
    public static ReferenceFacade getInstance()
    {
        return instance;
    }

    /**
     * Liefert eine Liste von Quellzuständen zum gegebenen Statuswechsel.
     *
     * @param stateChange Der Statuswechsel, repräsentiert durch ein Objekt vom Typ <code>StateChange</code>.
     */
}
```

```

* @return eine Liste von Quellzuständen zum gegebenen Statuswechsel
*
*/
public List<State> getSourceStates(final ReferenceStatemodel.StateChange stateChange)
{
    final List<State> lstSourceStates = new ArrayList<State>();

    switch (stateChange) {
        case PaymentReceived:
            // @replace Bitte bei Nuclet-Integration mit eigenem Code ersetzen!
            //
            // Beispiel:
            //
            // lstSourceStateIds.add(ProcessClientBillingSM.State_AB);
            // lstSourceStateIds.add(ProcessClientBillingSM.State_XY);

            break;
        default:
            break;
    }

    return lstSourceStates;
}

/**
* Liefert den Zielzustand zum gegebenen Statuswechsel.
*
* @param stateChange Der Statuswechsel, repräsentiert durch ein Objekt vom Typ <code>StateChange</code>.
*
* @return der Zielzustand von Quellzuständen zum gegebenen Statuswechsel
*
*/
public State getDestinationState(final ReferenceStatemodel.StateChange stateChange)
{
    switch (stateChange) {
        case PaymentReceived:
            // @replace Bitte bei Nuclet-Integration mit eigenem Code ersetzen!
            //
            // Beispiel:
            //
            // return ProcessClientBillingSM.State_ZZ;
            //

            return null;
        default:
            return null;
    }
}

/**
* Wrap the actual reference objects inside an <code>AbstractReferenceWrapper</code>
*
* @return the actual reference objects, wrapped inside an <code>AbstractReferenceWrapper</code>
*
*/
public AbstractReferenceWrapper getReferenceWrapper(final BusinessObject boReference)
{
    // @replace!
    //
    // return new ReferenceWrapper(boReference);

    return new ReferenceWrapper(boReference);
}

/**
* Get the class of the actual reference objects, i.e. the class wrapped by <code>ReferenceWrapper</code>
*
* @return the class of the actual reference objects, i.e. the class wrapped by <code>ReferenceWrapper<
/code>
*

```

```

    */
    public Class<T> getWrappedClass() // @replace Class<ClientBilling> getWrappedClass()
    {
        // @replace!
        //
        // return ClientBilling.class;

        return null;
    }

    /**
     * Get the attribute <code>NuclosState</code> of the actual reference objects
     *
     * @return the attribute <code>NuclosState</code> of the actual reference objects
     *
     */
    public PrimaryKeyAttribute<State> getNuclosStateAttribute()
    {
        // @replace! Bitte bei Nuclet-Integration mit eigenem Code ersetzen!
        //
        // Beispiel:
        //
        // ClientBilling.NuclosState;

        return null;
    }
}

```

Related pages

Nuclet: MT940 (not yet available)

MT940 Nuclet data sheet

[Open](#)

Interfaces

All integrations

[Open](#)