

4.7.3 CAMT: State change

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

📖 4.7.3 CAMT: State change

State change – part of the CAMT Nuclet documentation (import and processing of CAMT.053/CAMT.054 bank statements (Cash Management)).

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).

By default, the CAMT Nuclet provides for a state change on a reference object as soon as the conditions of payment are met. Since the reference entity is not part of the CAMT Nuclet, the source and destination states of this state change must be defined during integration.

More about checking the conditions of payment and the way the CAMT Nuclet works can be found in [section 2](#).

4.7.3.1 State change "PaymentReceived"

The state change described above is represented by the enum object "PaymentReceived" of type ReferenceStatemodel.StateChange. For this state change, source and destination states must be defined in the ReferenceFacade class.

State change	Type	Method	Return value	Note
PaymentReceived	Source states	getSourceStates()	List<State>	one or more source states can be defined
PaymentReceived	Destination state	getDestinationState()	State	the destination state must be defined unambiguously

The definition is necessary for the following reasons:

- During the potential state change, it is checked whether the objects are in a valid source state.
- When performing the state change, the concrete destination state must be specified.

Since the actual states from the state model used are not known in the MT940 Nuclet, this interface arrangement must be used.

```

/**
 * 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;
    }
}

```

Related pages

Nuclet: CAMT (available in the trade Nuclet)

CAMT Nuclet data sheet

[Open](#)

Interfaces

All integrations

[Open](#)

