Infor LN Customization Guide for External Integration Transactions

Important Notices

Trademark Acknowledgements

Publication Information

Document code	tfextinttransug (U9002)
Release	10.6 (10.6)
Publication date	March 26, 2018

Table of Contents

About this document

This document describes the process to set up the master data for the mapping scheme. The document also explains the process to extend the mapping scheme with the mapping setup for the integration document types, which are used in an external integration.

How to read this document

Comments?

We continually review and improve our documentation. Any remarks/requests for information concerning this document or topic are appreciated. Please e-mail your comments to documentation@infor.com .

In your e-mail, refer to the document number and title. More specific information will enable us to process feedback efficiently.

Contacting Infor

If you have questions about Infor products, go to the Infor Xtreme Support portal.

If we update this document after the product release, we will post the new version on this website. We recommend that you check this website periodically for updated documentation.

If you have comments about Infor documentation, contact documentation@infor.com .

Chapter 1
Introduction

1

Introduction

Purpose

In LN, Financials is open for financial integration transactions originating from external applications, such as a salary application.

This document describes the procedure that should be followed to achieve the eventual import and subsequent processing of external integration transactions in Financials.

Note

Logging of reconciliation transactions is not enabled for external integrations.

Definitions, acronyms and abbreviations

Definitions, acronyms and abbreviations used in the document are listed below in alphabetical order.

Term	Definition
DAL	Data access layer
DLL	Dynamic link library
GL code	Code that defines a specific combination of ledger account and dimensions
GUID	Globally unique identifier

Chapter 2
Procedure

2

Procedure

Before you can actually import integration transactions into Financials from an external source, you must first initialize a number of master data, and extend the mapping scheme with the mapping setup for the integration document types that are used in the external integration.

Master data

The integration mapping in LN Financials differs from the integration mapping in previous LN versions in many aspects. A major difference is that in LN, integration mapping makes use of a meta-modeled data model. Among other things, this means that database fields neither directly refer to specific objects, such as purchase order, service order, nor to specific elements, such as business partner, cost component, item. Instead, a meta model defines which business object and which element are referred to for a specific integration transaction (table tfgld482) or logged element (table tfgld481).

This meta-modeled master data must be initialized for use in the mapping scheme by an initialization script. For more information, refer to Chapter 3, "Master Data Setup."

Mapping scheme

After initialization of the external master data, create a new mapping scheme version (by copying an existing mapping scheme). This new mapping scheme version must be extended with the mapping setup for the external master data, that is, the new external integration document types. For the procedure, refer to Chapter 4. "Mapping Scheme Setup."

Integration transactions

After the new mapping scheme version has been activated, LN is ready to receive the new external integration transactions.

One of the most obvious methods to import external transactions into the Financials integration transactions tables, is to make use of an exchange scheme, as was done for previous LN versions.

Compared to previous LN versions, in some respects different data must be provided (including the business object and elements), and it is mandatory to log more data. For details, refer to Chapter 5, "External Integration Transactions."

Chapter 3
Master Data Setup

3

Master Data Setup

Initialization script

- An initialization script must be written which uses the DAL (DAL.NEW) of the involved tables to create rows in the database tables mentioned in the sections below.
- Naming conventions mentioned in this document should be strictly adhered to. Otherwise, mixing up with (future) standard data cannot be ruled out.

Business objects (tfgld454)

This table shows the business objects that will be part of integration transactions displayed in, for example, the Integration Transactions (tfgld4582m000) session.

Field	Description	Format	Domain	Mandatory *	Comments
bona	business object	str(17)	tcbona	Yes	must start with the "#"-character; other characters are free to choose; this name will be displayed in, for example, the Integration Trans-

					actions (tfgld4582m000)
lita	transaction logging allowed	enum	tcyesno	Yes	value: tcyesno.yes
maal	mapping allowed	enum	tcyesno	Yes	value: tcyesno.no

* For external data, this setting may be different from the table field's property.

Elements (tfgld456)

This table shows the elements that will be logged related to integration transactions, and will be primarily used for ledger account and dimension mapping. The elements are displayed when, for example, you set up the mapping scheme.

Field	Description	Format	Domain	Mandatory *	Comments
elcd	element code	str(12)	tfgld.elcd	Yes	must start with the string "EXT"; other characters must be numeric
bonm	business object	str(17)	tcbona	Yes	soft reference to tfgld454.bona
boat	business object attribute	str(17)	tcboat	Yes	must start with the "#" character; other characters are free to choose; this name will be displayed when you maintain the mapping scheme
elds	element description	str(30)	tcdsca	Yes	description of the element that will be visible to the user

levl	element level	byte	tfgld.srno	Yes	value: 1
pecd	parent element	str(12)	tfgld.elcd	No	should be left empty (not used externally)
rcel	reconciliation element	str(12)	tcrelc	No	should be left empty (not used externally)
bpyn	business partner	enum	tcyesno	Yes	value: tcyesno.no (not used externally)
expi	expired y/n	enum	tcyesno	Yes	value: tcyesno.no

* For external data, this setting may be different from the table field's property.

Integration document types (tfgld457)

This table shows the integration document types that will be part of integration transactions displayed in, for example, the Integration Transactions (tfgld4582m000) session.

Field	Description	Format	Domain	Mandatory *	Comments
idtc	integration document type	str(8)	tcidty	Yes	must start with the string "EXT"; other characters must be numeric
idtd	integration document type description	str(30)	tcdsca	Yes	must start with the "#" character; other characters are free to choose; this name will be displayed in, for example, the Integration Transactions

					(tfgld4582m000) session
bonm	business object	str(17)	tcbona	Yes	soft reference to tfgld454.bona
secd	sort element	str(12)	tfgld.elcd	no	soft reference to tfgld456.elcd; must only be filled if the sort element functionality is used
modt	modifiable document type	enum	tcyesno	Yes	value: tcyesno.no
ppdt	publish posting details	enum	tcyesno	Yes	value: tcyesno.no (not used externally)
expi	expired y/n	enum	tcyesno	Yes	value: tcyesno.no
lemm	ledger mapping mandatory	enum	tcyesno	Yes	value: tcyesno.yes
dlae	log all elements debit	enum	tcyesno	Yes	value: tcyesno.yes
clae	log all elements credit	enum	tcyesno	Yes	value: tcyesno.yes
maal	mapping allowed	enum	tcyesno	Yes	value: tcyesno.yes
ltia	linked to integration account	enum	tcyesno	Yes	value: tcyesno.yes

* For external data, this setting may be different from the table field’s property.

Elements by integration document type (tfgld458)

This table shows the elements that will be logged for a particular integration document type, and that can be used within the mapping scheme for ledger account and dimension mapping.

Field	Description	Format	Domain	Mandatory *	Comments
idtc	integration document type	str(8)	tcidty	Yes	refers to tfgld457.idtc
elcd	element code	str(12)	tfgld.elcd	Yes	refers to tfgld456.elcd
dele	debit ledger mapping applicable	enum	tcyesno	Yes	value: tcyesno.yes
crle	credit ledger mapping applicable	enum	tcyesno	Yes	value: tcyesno.yes
nefi	never filled	enum	tcyesno	Yes	value: tcyesno.no
bpgl	business partner for general ledger	enum	tcyesno	Yes	value: tcyesno.no (not used externally)

* For external data, this setting may be different from the table field's property.

Additional procedure steps

After running the initialization script, complete these steps:

1. To update the business object and integration document type language dependent descriptions, in the Integration Parameters (tfgld4150s000) session, from the appropriate menu, select **Initialize Descriptions**.
2. In the Period Handling by Integration Document Type (tfgld4579m000) session, for each external integration document type – debit/credit combination, set **Closed Period Handling** to either **Post to next open period** or **Post to current period**. (Exception handling is only applicable for internal business objects.)

   Because table tfgld479 must be shared for all financial companies (please check!), you do not need to repeat this step for each individual financial company.

Chapter 4
Mapping Scheme Setup

4

Mapping Scheme Setup

Procedure

Next, set up the ledger account and dimension mapping for the newly added external integration document types, using the standard mapping scheme functionality.

This should be done according to the company's usual procedure for adding new mappings to the mapping scheme. The most obvious way to do this is to copy the existing mapping scheme to a new mapping scheme version. Subsequently, you can add the mapping for the external document types. Finally, the mapping scheme consistency must be checked, and, after all errors have been corrected, you can activate the mapping scheme.

Chapter 5
External Integration Transactions

5

External Integration Transactions

Now, the system is ready to receive financial integration transactions from an external source.

One of the most obvious methods to import external transactions into the Financials integration transactions tables, is to make use of an exchange scheme, as was done for previous LN versions.

The following sections list the requirements for the logged transactions.

These database tables are involved:

- Integration transactions (tfgld482)
- Logged elements (tfgld481)
- Unmapped/unposted integration transactions (tfgld487)
- GL code by integration transaction (tfgld486), if GL codes are involved

Integration transactions must be logged in pairs (in tfgld482): a credit transaction and an accompanying debit transaction. Both transactions will be stored in the same financial company, which is the financial company of the credit transaction (this is important for intercompany postings). Both transactions must be assigned the same GUID (see below). The transactions will refer to the same unmapped/unposted data in (tfgld487).

Each integration transaction can have a set of related logged elements (tfgld481). If the integration transaction has a GL code attached, an additional row must be stored in (tfgld486).

If a GL code is used, the mapping scheme will be bypassed in determining ledger account and dimensions, and the transactions will be posted to the ledger account and dimensions defined for the GL code.

Integration transactions (tfgld482)

This table contains the individual integration transactions.

Note

Related debit and credit transactions must have the same GUID assigned.

Field	Description	Format	Domain	Mandatory *	Comments
guid	globally unique identifier	str(22)	tcguid	Yes	must be the same for related debit and credit transactions; to be assigned using function uuid.generate$()
dbcr	debit/credit indicator	enum	tfgld.dbcr	Yes	
ocmp	source logistic company	int(3)	tcncmp	Yes	logistic company in which the business object resides; must be the same for related debit and credit transactions
idtc	integration document type	str(8)	tcidty	Yes	must be the same for related debit and credit transactions; refers to tfgld457.idtc
trdt	transaction date	utc date	tcdate	Yes	date on which, among others, the document date will be based; must be the same for related debit and credit transactions

tcmp	logistic company	int(3)	tcncmp	Yes	same as source logistic company (tfgld482.ocmp)
secd	sort element	str(12)	tfgld.elcd	No	soft reference to tfgld456.elcd; must only be filled if the sort element functionality is used
seva	sort element value	str(50)	tcelva	No	belongs to tfgld482.secd
rbon	business object	str(17)	tcbona	Yes	must be the same for related debit and credit transactions; refers to tfgld454.bona
rbid	business object ID	str(11)	tcboid	Yes	must be the same for related debit and credit transactions; the ID of the business object, for example, the purchase order number in case of a purchase order
rpon	sort position	long(8)	tcsopo	No	must be the same for related debit and credit transactions; used in indices for detailed sorting by business object (see indices tfgld482), among others, used in Integration Transac-

					tions (tfgld4582m000)
obre	business object reference	str(40)	tcborf	No	must be the same for related debit and credit transactions; extension of the business object ID, to specify more precisely what origin the transaction relates to; first part should be the same as tfgld482.rpon
buid	business object reference GUID	str(22)	tcguid	No	leave empty: not used externally
bpid	business partner	str(9)	tccom.bpid	No	soft reference to tccom100; business partner that should be logged with the transaction in the General Ledger
ttyp	transaction type	str(3)	tfgld.ttyp	No	leave empty; determined during posting
docn	document number	long(8)	tfgld.docn	No	leave empty; determined during posting
btno	batch number	long(6)	tfgld.btno	No	leave empty; determined during posting
lino	line number	long(8)	tfgld.lino	No	leave empty; determined during posting

sint	transaction status	enum	tfgld.ints	Yes	must be set to tfgld.ints.mapped if a GL code is involved, otherwise must be defaulted to tfgld.ints.logged
crdt	creation date	utc date	tcdate	Yes	must be the same for related debit and credit transactions; the date of the actual creation of the integration transaction
crus	user (creation)	str(16)	tfgld.user	Yes	user ID (logname$) of the user who logs the transaction; must be the same for related debit and credit transactions
usgr	financial user group	str(6)	tfgld.usgr	No	refers to tfgld435; to be retrieved from tfgld436 for tfgld482.crus; however, does not need to exist
pous	user (posting)	str(16)	tfgld.user	No	leave empty; determined during posting
podt	posting date	utc date	tcdate	No	leave empty; determined during posting
prin	printed	enum	tcyesno	Yes	value: tcyesno.no

glcu	GL code used	enum	tcyesno	Yes	value must be set to tcyesno.yes if a GL code is used, otherwise tcyesno.no
fcom	financial company	int(3)	tcncmp	Yes	financial company to whose General Ledger the transaction must be posted; can differ between the credit transaction and the debit transaction it belongs to; for the credit transaction, it must always be the financial company in which the integration transactions are logged (the physical financial company)
leac	ledger account	str(12)	tfgld.leac	No	must be empty if no GL code is used; otherwise, it must be filled by retrieval from tfgld475
dim1	dimension 1	str(6)	tfgld.dimx	No	see tfgld482.leac
dim2	dimension 2	str(6)	tfgld.dimx	No	see tfgld482.leac
dim3	dimension 3	str(6)	tfgld.dimx	No	see tfgld482.leac

dim4	dimension 4	str(6)	tfgld.dimx	No	see tfgld482.leac
dim5	dimension 5	str(6)	tfgld.dimx	No	see tfgld482.leac
amnt	transaction amount	double	tfgld.amnt	Yes	must be the same for related debit and credit transactions; must be properly rounded; zero can be a valid value
ccur	transaction currency code	str(3)	tcccur	Yes	must be the same for related debit and credit transactions; refers to tcmcs002
ratd	currency rate date	utc date	tcdate	Yes	date for which the currency rates towards the home currencies have been retrieved; probably equal to tfgld482.trdt
rtyp	currency rate type	str(3)	tcrtyp	Yes	rate type used in the home amounts calculation; will probably be equal to the (financial) company's internal rate type (tcemm170.exeu)
rate	currency rates	double (array)	tcratc	Yes	rates towards the transaction's financial company's home cur-

					rencies; can be retrieved using the standard tcemm.dll5000/5010 functions
ratf	currency rate factors	long(6) (array)	tcratf	Yes	rate factors towards the transaction's financial company's home currencies; can be retrieved using the standard tcemm.dll5000/5010 functions
eibc	express in base currency indicators	enum (array)	tcyesno	Yes	**Express in Base Currency** indicators towards the transaction's financial company's home currencies; can be retrieved using the standard tcemm.dll5000/5010 functions
amth	amounts in home currency	double (array)	tfgld.amnt	Yes	amounts in the transaction's financial company's home currencies; can be calculated using the standard tcemm.dll5000/5010 functions; must be properly rounded after calculation
cuni	the quantity's unit	str(3)	tccuni	No	refers to tcmcs001; must be

					filled if a quantity is logged
nuni	quantity	double	tfgld.quan	No	belongs to tfgld482.cuni
fyer	financial year	int(4)	tfgld.year	Yes	see tfgld482.fprd
fprd	financial period	int(2)	tfgld.prod	Yes	must be filled, and must exist in tfgld005; can be determined based on tfgld482.trdt (transaction date and financial period should preferably be in line)
ryer	reporting year	int(4)	tfgld.year	No	mandatory if reporting periods are used; see tfgld482.fyer
rprd	reporting period	int(2)	tfgld.prod	No	mandatory if reporting periods are used; see tfgld482.fprd
tyer	tax year	int(4)	tfgld.year	Yes	see tfgld482.fyer
tprd	tax period	int(2)	tfgld.prod	Yes	see tfgld482.fprd
mscd	used mapping scheme code	str(6)	tfgld.mscd	No	leave empty; determined during mapping
msvs	used mapping scheme version	long(5)	tfgld.msvs	No	leave empty; determined during mapping

* For external data, this setting may be different from the table field's property.

Logged elements (tfgld481)

This table contains the logged elements by integration transaction, which elements will be used for mapping of the integration transaction.

Field	Description	Format	Domain	Mandatory *	Comments
guid	globally unique identifier	str(22)	tcguid	Yes	soft reference to tfgld482.guid
dbcr	debit/credit indicator	enum	tfgld.dbcr	Yes	soft reference to tfgld482.dbcr
seqn	sequence number	int(4)	tcpono	Yes	sequence number to extend the primary key; initial 30 elements must be stored in row with sequence number 1, next 30 elements in row number 2, and so on.
elcd	element codes	str(12) (array)	tfgld.elcd	Yes	each element (softly) refers to tfgld456.elcd; elements with empty values do not need to be logged
elva	element values	str(50)	tcelva	Yes	belongs to tfgld481.elcd

* For external data, this setting may be different from the table field's property.

Unmapped/unposted integration transactions (tfgld487)

This table contains the integration transactions pairs, and will be used during mapping and posting. This table is a parent table of tfgld482.

Field	Description	Format	Domain	Mandatory *	Comments
guid	globally unique identifier	str(22)	tcguid	Yes	= tfgld482.guid
sint	transaction status	enum	tfgld.ints	Yes	value: tfgld.ints.logged
ocmp	source logistic company	int(3)	tcncmp	Yes	= tfgld482.ocmp
fcom	credit financial company	int(3)	tcncmp	Yes	the credit transaction's financial company
rbon	business object	str(17)	tcbona	Yes	= tfgld482.rbon
idtc	integration document type	str(8)	tcidty	Yes	= tfgld482.idtc
trdt	transaction date	utc date	tcdate	Yes	= tfgld482.trdt
usgr	financial user group	str(6)	tfgld.usgr	No	= tfgld482.usgr
crdt	creation date	utc date	tcdate	Yes	= tfgld482.crdt

* For external data, this setting may be different from the table field's property.

GL code by integration transaction (tfgld486)

This table shows the GL code that was used in an integration transaction.

Field	Description	Format	Domain	Mandatory *	Comments
guid	globally unique identifier	str(22)	tcguid	Yes	= tfgld482.guid
dbcr	debit/credit indicator	enum	tfgld.dbcr	Yes	= tfgld482.dbcr
fcom	financial company	int(3)	tcncmp	Yes	= tfgld482.fcom; soft reference to tfgld475.fcmp
glco	GL code	str(50)	tcglco	Yes	soft reference to tfgld475.glcd

* For external data, this setting may be different from the table field’s property.