

Upgrade Guide | PUBLIC
Software Update Manager 2.0 SPS 24
Document Version: 1.0 – 2025-10-06

Downtime-Optimized Conversion
To SAP S/4HANA

© 2025 SAP SE or an SAP affiliate company. All rights reserved.

THE BEST RUN SAP

Content

1 Document History. 4

2 Introduction. 5

2.1 About This Document. 5

2.2 Integration/Further Documentation. 5

2.3 Naming Conventions. 6

2.4 Basic Idea of Downtime-Optimized Conversion to SAP S/4HANA 6

2.5 Prerequisites and Restrictions. 6

3 Planning. 8

3.1 Downtime and Business Restrictions. 8

3.2 Customizing Freeze. 9

3.3 Phases Specific to Downtime-Optimized Conversion. 22

3.4 Database Triggers. 23

3.5 Virtualization and Isolation of Clone due to Delta Loads Verfication. 24

3.6 Table Partitioning. 24

3.7 Shadow Field Handling. 24

3.8 Homogeneous Database Migration Option for SAP ERP on SAP HANA. 26

3.9 Software Update Manager Toolbox. 26

3.10 Specific Phases Having a Long Runtime or Resulting in a Timeout 26

4 Preparation. 28

4.1 Relevant SAP Notes. 28

4.2 Preparation of Software Update Manager. 29

4.3 Project Sequence for the Downtime-Optimized Conversion. 29

4.4 Prepare Your Financial Data for the Downtime-Optimized Conversion. 31

4.5 Preparation of Database Migration Option. 31

4.6 Test Data for SAP S/4HANA Finance During the Test Run. 32

4.7 Moving Long-Running XCLA Processing to the Uptime. 32

4.8 Considerations for Long-Running Batch Jobs in Phase “RUN_WAIT_FOR_BATCHES”. 32

5 Roadmap Steps with Downtime-Optimized Conversion Features. 33

5.1 Configuration. 35

5.2 Preprocessing. 35

5.3 Execution. 36

6 Follow-Up Activities. 37

6.1 Performing IMG Steps After the Conversion. 37

6.2 Performing Further Follow-Up Activities. 38

7 Troubleshooting. 39

7.1 Generic Troubleshooting Activities. 39

Finding the Relevant Log Files. 39

Analyzing Short Dumps. 39

Checking the Job Overview. 40

Checking the Database Logs. 40

Checking the DBA Cockpit. 40

Oracle Only: Deprecated Datatypes 41

Handling Function Call Failure. 41

7.2 Troubleshooting for the Finance and Material Management-Inventory Management Migration 41

Finance Migration. 41

MM-IM. 42

7.3 Performance Impact. 43

7.4 Monitoring the Replication. 43

7.5 Correcting Errors in Specific Phases. 44

Handling Missing Partitioning in Phase MAIN_MIGOPT/SUBMOD_SMIG_PREP_DMO/ JOB_FINS_UPG_MIG_PREPARE. 44

Handling Errors in Phase MAIN_MIGOPT/SUBMOD_SMIG_PREP_DMO/ JOB_FINS_UPG_MIG_PREPARE. 44

Handling Errors in Phase SMIG_SFIN_MIGPREP_TRC_EVA. 45

Handling Errors in Phase REQ_PARTITIONING_MATDOC. 46

Handling Errors in Phase PARRUN_SMIG_SFIN. 46

Handling Short Dumps in Phase PARRUN_SMIG_UT_MKPF1_S4 48

Handling ST22 Dump due to Freeze Trigger. 49

Handling Errors During Uptime Replication. 49

Handling DBSL Error 99 in Phase SCEXEC_GRANT_RSV_DT. 50

Handling Errors in Phase MAIN_NEWBAS/PARMVNT_UPG. 50

Handling Missing Database Objects in Phase MAIN_POSTPROC/RUN_RSDB02CK_END_DMO 50

1 Document History

The following table provides an overview of the most important document changes.

> ⚠ Caution
>
> Before you start, make sure that you have the latest version of this document. You can find the latest version as attachment to SAP Note 3619822.

Document: Downtime-Optimized Conversion to SAP S/4HANA

Version	Date	Description
1.0	2025-10-06	Initial version

2 Introduction

2.1 About This Document

This guide gives you an overview of the downtime-optimized conversion procedure to SAP S/4HANA using the Software Update Manager (SUM).

The document includes special planning, preparation, and follow-up activities as well as information about the downtime-optimized conversion procedure to S/4HANA. It also provides you with knowledge that you need when you encounter errors.

The information in this document is intended for SAP system administrators with operating system, database, and SAP NetWeaver Application Server knowledge.

2.2 Integration/Further Documentation

In addition to this document, you require the following further documents:

Document	Title
SUM Guide	*Updating SAP ABAP Systems on <operating system>: <database>* This *SUM guide* applies to the Software Update Manager **2.0**. It describes how to update SAP systems based on SAP NetWeaver using the Software Update Manager (SUM).
DMO Guide for target database SAP HANA	*Database Migration Option: Target Database SAP HANA* This *DMO guide* applies to the Software Update Manager **2.0**. It describes all the activities that you need to do for a database migration.

The latest version of the guides are available on the SAP Support Portal at: http://support.sap.com/sltoolset *tab* > *System Maintenance* .

See also the required SAP Notes in Relevant SAP Notes [page 28].

2.3 Naming Conventions

This section deals with the most important naming conventions used in this guide.

See the *SUM Guide* for a comprehensive overview about naming conventions. The following table describes the most important naming conventions in this guide:

Naming	Meaning
System conversion	Overall process of the transfer of an existing SAP ERP system to SAP S/4HANA system
Data conversion	Conversion of data from the previous data structure to the new SAP S/4HANA data structure
Downtime-optimized conversion	Procedure to reduce the downtime for the data conversion as a part of the system conversion
Migration	Transfer of database content from the source database to a target database such as SAP HANA
SAPup	Executable that is used as a part of SUM

2.4 Basic Idea of Downtime-Optimized Conversion to SAP S/4HANA

You can update and migrate your SAP ECC system to SAP S/4HANA in just one procedure and significantly reduce your business downtime. The downtime-optimized conversion procedure conducts a part of the data conversion in the uptime processing, which significantly reduces the overall business downtime.

The downtime-optimized conversion feature is delivered with the Software Update Manager tool as a part of the Software Logistics Toolset (SL Toolset).

The main benefit of downtime-optimized conversion is to reduce the technical downtime.

2.5 Prerequisites and Restrictions

The downtime-optimized conversion requires specific project planning as it has limitations and restrictions that must be considered during the project planning.

See SAP Note 3619822 for more information about the following:

- Supported target product versions
- Required SAP HANA database versions

- Different restrictions for the downtime-optimized conversion to SAP S/4HANA

For more planning considerations and activities, see chapter *Planning* of this guide.

> **Note**
>
> Consider that the downtime-optimized conversion can have a **long overall runtime**. This is due to the large data volume of the conversion to SAP S/4HANA and due to the data inconsistencies that occur in FI checks and FI conversion because of historical data.

3 Planning

This part of the guide contains information about planning your SAP system update and the downtime-optimized conversion procedure.

3.1 Downtime and Business Restrictions

To obtain a transactional consistent state of `Asset Accounting (FI-AA)` before cloning the tables, new postings in `Asset Accounting` must be temporarily locked. By freezing the leading `Asset Accounting` tables in your system, we ensure that no further postings can be generated in the system (except for the postings required for period-end closing). The `Asset Accounting` lock is removed in the `REQ_SMIG_APPL_UNLOCK` phase.

> ⚠ Caution
>
> Additionally, starting with the `Asset Accounting` lock, the Software Update Manager locks the following transactions in the source system till the end of the downtime::
>
> - SARA
> - KA12
> - KA16
> - AS91
> - AJRW
> - AJAB

Consider also the following constraints for the Finance components when planning the downtime:

- For any `Accounting` component during the delta period (time after the short term lock for `Asset Accounting` until entering the downtime; time when the database triggers are active):
  - Do not reopen fiscal periods that have already been closed and do not post to fiscal years that have already been carried forward. To reduce the delta conversion runtime, we recommend that you minimize the number of fiscal periods that are posted to ideally just the current fiscal period.
  - Do not perform any archiving.
  - Do not perform any balance carryforward.
  - Do not change anything in the Customizing.
  - Do not change anything in the repository.
- Asset Accounting (FI-AA)
  - Do not transfer legacy data (for example, using transaction AS91).
  - Do not carry out any fiscal year-end closing activities in `Asset Accounting` using transactions `AJRW` and `AJAB`.
  - Do not adjust any `Asset Accounting` data using reports that have names beginning with `RACORR*`.
  - Do not reopen any closed fiscal years.

- Do not post full retirements or full retiring transfers starting the first day of the open period when triggers are activated.
- Do not change *Depreciation Key, Expired Useful Life* or *Depreciation Start Date* of an asset during the delta phase.
- To minimize the data volume and the downtime, avoid performing full regular period depreciation posting runs for all assets during the delta phase (calling report RAPOST200 and transaction **AFAB**).

- Material Ledger Accounting
  - If you have activated `Actual Costing` in your system before the system conversion to SAP S/4HANA: Complete all `Material Ledger` costing runs for actual costing (transaction `CKMLCP`) as well as alternative valuation runs (transaction `CKMLCPAVR`). The reason for this is that after the conversion, you will not be able to perform any modifications to costing runs created before the conversion.
  - Do not change `Material Ledger` costing runs and do not run steps of `Material Ledger` costing runs during the conversion.
  - Do not activate or deactivate `Material Ledger Actual Costing` during the conversion for one or more plants.
    For more information, see SAP Note 2354768.
  - Do not change the exchange rates for the date of the conversion when triggers are active.
  - Do not use transaction `CKMSTART` (*Production Startup with Material Ledger*).
  - Do not use transaction `MMPI` (*Initialize Period for Material Master Records*).
- Inventory Management (MM-IM)
  - In the source system, do not archive materials and material documents (object `MM_MATBEL`).
  - Do not start `MM-IM` inconsistency reports to correct `MM-IM` data. Such changes are recorded in table `MMINKON_UP`.
  - Do not change the base unit of measure `MARA-MEINS` to simplify the conversion procedure.

3.2 Customizing Freeze

Customizing Freeze initiates a phase during which Customizing changes as well as many Finance and Controlling activities are no longer possible.

The Customizing Freeze starts once the production system is copied to the sandbox to prepare the final Finance Customizing transport. From this point in time, the Finance Customizing must not be changed in the production landscape. New Finance and Controlling-related Customizing such as new company codes would require additional Finance Customizing for the conversion. In order to get the Finance Customizing request, the sandbox run must be repeated in this case.

The Hard Freeze starts during the uptime execution for the Software Update Manager. It comes with additional restrictions that are listed in section *Downtime and Business Restrictions* and in SAP Note 3619822.

> **Note**
>
> Note that transaction SPRO must not be used in the shadow/temporary system although it is not locked. Instead, you may use the transaction UPG_SFIN_CUST on the shadow/temporary system to do additional changes that were made after the inclusion of the Customizing request. There is only a very limited scope of Finance Customizing that can be changed during a downtime-optimized procedure run. After SUM is finished, transaction SPRO is allowed to be used again.

The following tables display which Finance and Controlling activities are restricted or not possible (create, change, delete), once the Customizing Freeze has started:

Finance

- **Organizational Structure**

Object	Create	Change	Delete	Details
Company Code	No	No	No	The conversion can be stopped because of missing Customizig that was not identified in previous test conversions. Settings which are not compatible with New General Ledger can create inconsistencies.
Company Code Address	Yes	Yes	No	A deletion will cause inconsistencies in Customizing.
Company Code Global Parameter	No	No	No	Very critical settings for the migration of line items from BSEG to ACDOCA
Posting Period	No	No	No	Very critical settings for the migration of line items from BSEG to ACDOCA
Fiscal Year Variant	No	No	No	Very critical settings for the migration of line items from BSEG to ACDOCA
Chart of Accounts	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger

Object	Create	Change	Delete	Details
Segment	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated
Functional Area	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated
Business Area	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated
Trading Partner	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated
Currency	No	No	No	A change of the currency setting will cause inconcistencies in BSEG and are not tested during the test conversions
Chart of Accounts Assignment to CC	No	No	No	The change will cause inconsistencies in BSEG which will stop the MIG-Cockpit during the data conversion
Company Code Assignment to Fiscal Year	No	No	No	The change will cause inconsistencies in BSEG which will stop the MIG-Cockpit during the data conversion

- **Master Data**

Object	Create	Change	Delete	Details
General Ledger Account Description	Yes	Yes	Yes	Not relevant for migration
Profit Center	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Please consider that also the historical data will be migrated
General Ledger Account Control Data	No	No	No	The change of control data will cause inconsistencies in master data and BSEG and stop the MIG-Cockpit
General Ledger Account (description, bank details, interest calculation)	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated
Reconciliation Account	Yes	No	No	A change or deletion of Reconciliation accounts for Fixed Assets and Materials could cause additional unexpected error messages during migration. Reconciliation accounts for customer/vendor are not technically used during migration, but a change of these might cause unexpected errors in preparation activities (SAPF190 or TFC_COMPARE_VZ).
Open Item Managed Account	Yes	Yes	No	A change or deletion of open item managed accounts, or of the property `<open item managed>`, will cause inconsistencies in Master Data and Transactional Data tables.

- **General Ledger**

Object	Create	Change	Delete	Details
Payment Terms	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated.
Tax settings (Withholding Tax, electronic VAT Report, Tax Code)	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated.
Account Determination	Yes	Maybe	No	The migration cannot be executed if a value from source system is not existing in the target system. Please consider that also the historical data will be migrated. A changed account determination that is used for ML or FI-AA postings during migration can lead to errors if it hasn't been tested before (transactions OBYC and AO90). Changes to the CO default account assignment (OKB9) are not critical.
Document Types	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated.

Object	Create	Change	Delete	Details
Number Ranges	Yes	Maybe	No	The migration cannot be executed if a value from source system is not existing in the target system. Please consider that also the historical data will be migrated. A change of the current number of a used number range can be critical.
Field Status Group	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated.
Foreign Currency Valuation	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger
Custom Fields	No	No	No	The migration cannot be executed correctly if you have custom fields for which customizing was not performed
Data Migration Balances (P&L, BS)	No	No	No	High risk that during data migration (via idoc, database migration tools) new inconsistencies will be created in database

- Asset Accounting

Object	Create	Change	Delete	Details
Chart of Depreciation	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger

Object	Create	Change	Delete	Details
Depreciation Area	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger
Document Types	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated
Number Ranges	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated
Chart of Depreciation assignment to CC	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger
Asset Class	Yes	No	No	Very high risk that a made setting can be incompatible with the New Asset Accounting and New General Ledger. Creation of new asset classes is not critical unless they are already being used.
Account Determination	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger
Integration with General Ledger	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger

Object	Create	Change	Delete	Details
Depreciation Calculation Rules	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger
Currency	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger
Data Migration - Asset Values	No	No	No	Very high risk for a made setting to be incompatible with the New Asset Accounting and New General Ledger

- **Bank Accounting**

Object	Create	Change	Delete	Details
Bank Accounts	Yes	Maybe	Maybe	Typical changes are not critical for migration. Additionally, there is an option to start the migration of house bank accounts after the technical upgrade from IMG.. For more information, see IMG Activity *Switch Migration Program for House Bank Accounts*, Tx SM30 FCML_BAM_MIGCONF

Object	Create	Change	Delete	Details
House Banks	Yes	Maybe	Maybe	Typical changes are not critical for migration. Additionally, there is an option to start the migration of house bank accounts after the technical upgrade from IMG.. For more information, see IMG Activity *Switch Migration Program for House Bank Accounts*, Tx SM30 FCML_BAM_MIGCONF
Payment Files	Yes	Yes	Yes	Not relevant for migration
Payment Methods	Yes	Yes	Yes	Not relevant for migration
Forms	Yes	Yes	Yes	Not relevant for migration
Electronic Bank Statement	Yes	Yes	Yes	Not relevant for migration
Payment Run	Yes	Yes	Yes	Not relevant for migration
Payment Advice	Yes	Yes	Yes	Not relevant for migration
Dunning	Yes	Yes	Yes	Not relevant for migration

- **Accoun Payables or Account Receivables**

Object	Create	Change	Delete	Details
Forms	Yes	Yes	Yes	Not relevant for migration
Enjoy Transactions	Yes	Yes	Yes	Not relevant for migration
SD/MM Account Determination	Yes	Yes	Yes	Not relevant for migration
Regrouping	Yes	Yes	Yes	Not relevant for migration
Dunning	Yes	Yes	Yes	Not relevant for migration

Controlling

- Organizational Structure

Object	Create	Change	Delete	Details
CC Assignment to Controlling Area	No	No	No	A change causes inconsistencies in BSEG which will stop the MIG-Cockpit during the data conversion
Currency	No	No	No	A change of the currency setting causes inconcistencies in BSEG and are not tested during the test conversions.
Ledger for CO Versions	No	No	No	A change causes inconsistencies in BSEG which will stop the MIG-Cockpit during the data conversion
Profitability Analysis	No	No	No	A change causes inconsistencies in BSEG which will stop the MIG-Cockpit during the data conversion
Operating Concern	No	No	No	A change causes inconsistencies in BSEG which will stop the MIG-Cockpit during the data conversion
Controlling Area	No	No	No	A change causes inconsistencies in BSEG which will stop the MIG-Cockpit during the data conversion

- Master Data

Object	Create	Change	Delete	Details
Cost Center	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated.

Object	Create	Change	Delete	Details
Internal Order	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated.
WBS Elements	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated.
Cost Elements	Yes	Yes	No	The migration cannot be executed if a value from source system is not existing in the target system. Consider that also the historical data will be migrated.

- **General Controlling**

Object	Create	Change	Delete	Details
Custom Fields	No	No	No	The migration cannot be executed correctly if you have custom fields for which customizing was not performed
Profitability Segments	No	No	No	You must not change the operating concern structure in Tx KEA0. If you already use account-based CO-PA you must not run correction reports that update profitability segments in the CE4* tables
Operating Concern	No	No	No	The settings on operating concern should be the same between source and target system (or you will data inconsistancies)

Object	Create	Change	Delete	Details
Profitability Analysis	Maybe	Maybe	Maybe	Changes to settings like derivation rules are not critical for migration. But be careful if such settings are usually maintained in production. IMG activity *Transport operating concern*((Tx KE3I) is part of *Preparations and Customizing for Conrolling* - make sure to transport basic settings only to not overwrite derivation rules or other settings in production accidentally.

- **Material Ledger**

Object	Create	Change	Delete	Details
Material Ledger and Valuation Settings	No	No	No	The conversion can be stopped because of missing Customizing that was not identified in previous test conversions. Settings may not be compatible with Material Ledger. New inconsistencies might be created in database because of missing Customizing.

Object	Create	Change	Delete	Details
Valuation Area / Plants	Maybe	No	No	The conversion can be stopped because of missing Customizing that was not identified in previous test conversions. Settings may not be compatible with Material Ledger. New inconsistencies might be created in database because of missing Customizing. It is technically possible, however, to create missing customizing in the temporary instance using the IMG activities decribed under *Preparation and Migration of Customizing for Material Ledger*
Valuation Level	No	No	No	The conversion can be stopped because of missing Customizing that was not identified in previous test conversions. Settings may not be compatible with Material Ledger. New inconsistencies might be created in database because of missing Customizing.

- **Product Cost Controlling**

Object	Create	Change	Delete	Details
Cost Component Structure	No	No	No	Cost Component Structure must not be changed. Assigning new cost elements to existing cost component structure is always critical and not recommended at all.

3.3 Phases Specific to Downtime-Optimized Conversion

The downtime-optimized conversion has additional phases.

In addition to the phases of a standard conversion run, the following SUM phases are specific to the downtime-optimized conversion:

Classification of downtime-optimized procedure, downtime-optimized Database Migration Option, and table classification:

- `RUN_CLASSIFY_DODMO → downtime-optimized Database Migration Option`
- `RUN_RSPTBFIL_NZDM_CLASSIFY → near-Zero Downtime Maintenance (phase runs for source systems on SAP HANA DB)`

Activation of triggers for recording and replay:

- `MAIN_SHDRUN/SUBMOD_RRC_TRIGGER_DEFAULT/SQLRUNTASK_SWITCH_RRC_TRIGGER_MODE_UKR`

Short term lock for Asset Accounting:

- `MAIN_SHDRUN/SUBMOD_SMIG_APPL_FREEZE/REQ_SMIG_APPL_LOCK` → Begin of lock
- `MAIN_SHDRUN/SUBMOD_SMIG_APPL_UNFREEZE/REQ_SMIG_APPL_UNLOCK` → End of lock

Uptime replication of recorded changes:

- `SUBMOD_REPLAY_RRC_START/PROCESS_REPLICATE_RRC_START/PROCESS_REPLICATE_RRC`

Finance uptime migration:

- .../SUBMOD_SMIG_SCENARIOS/SUBMOD_SMIG_SFIN/PARRUN_SMIG_SFIN

MM-IM uptime migration:

- `.../SUBMOD_SMIG_UT_MMIM/PARRUN_SMIG_UT_MKPF1_S4`
- `.../SUBMOD_SMIG_UT_MMIM/PARRUN_SMIG_UT_MKPF2_S4`
- `.../SUBMOD_SMIG_UT_MMIM/RUN_SMIG_UT_MKPF3_S4`

Final replication in downtime:

`.../SUBMOD_RRC_FINAL/RUN_RRC_REPLICATE_FINAL`

Finance downtime "delta" migration:

- `.../SUBMOD_SMIG_SCENARIOS/SUBMOD_SMIG_SFIN/PARRUN_SMIG_SFIN`

MM-IM downtime "delta" migration:

- `…/SUBMOD_SMIG_SCENARIOS/ SUBMOD_SMIG_DT_MMIM/*`

3.4 Database Triggers

The downtime-optimized conversion approach migrates and converts application data already during the uptime.

As the table content is potentially changed by end user activity after the initial migration and conversion, those changes must be considered by the SUM. Therefore, the Change Record and Replay (CRR) technology is built into the SUM that is based on database triggers. Database triggers define actions that are executed when the corresponding database table is changed. The database trigger will write respective information into trigger log tables that allow a later replication of the changes (Delta Replication) to the target. The respective logging tables and all other tables associated to this process are listed below.

Description	Purpose	Technical ID	SUM Phase
LoggingTable	Records changes of business operations until it has been replayed	1CRR/LTX<infoID> **For Oracle and SAP HANA Only:** /1CRR/LTY<infoID>* (For more details, see the *information below the table.)	`RUNASYN_RRC_REPLICATE`
LoggingTable History	Holds processed entries of logging table	/1CRR/LTH<infoID>	`RUNASYN_RRC_REPLICATE`
Snapshot Table (Transparent | Cluster)	Contains information in order to“travel back in time“ and to create the state of T0 (transactional consistent state) for tables used in UT Conversion	/1CRR/RSC<infoID> /1CRR/R<infoID>	`RUN_RRC_REVERT2SNAPSHOT`
Snapshot Table History	Holds processed entries of snapshot tables for reasons of traceability	/1CRR/BSC<infoID> /1CRR/B<infoID>	`RUN_RRC_REVERT2SNAPSHOT`
Protocol Table	Contains information about the effective changes since T0 and is needed during the downtime-optimized conversion	/1CRR/PRT<infoID>	`RUN_SMIG_FILL_PROTTAB PARRUN_SMIG_SFIN`
Backup Table	Handles possible clash for table with delivery class G which might be changed by production operation as well as the conversion	/1CRR/BCK<infoID>	`RUNASYN_RRC_REPLICATE`

For Oracle and SAP HANA Only: The Software Update Manager creates two logging tables for each application table. The first logging table saves up to five million changed records. This is the default threshold value. When this threshold is reached, the second logging table takes over. If the threshold is reached again, the first logging table then takes over, and so on. This ensures the size of single logging table does not grow to an unmanageable size.

> **Note**
>
> <infoID> is the unique identifier number for each table which is replicated. You can find it in table `PUTTB_SHD` in field `INFOID` of the recorded table.

3.5 Virtualization and Isolation of Clone due to Delta Loads Verfication

The concept of Delta Load Verification aims to simulate the downtime of the approach on production-like hardware. It creates a clone of the production system at the time of the downtime dialog. This only works if the clone is created using virtualization, so that the clone is identical from perspective of SUM (e.g. same host names). In addition, the clone must be completely isolated from the network of the production system.

3.6 Table Partitioning

As a system conversion generally starts in an SAP ECC system on at least SAP NetWeaver AS ABAP 7.0 or a higher version, automatic partitioning of large tables is provided for the downtime-optimized conversion run.

In addition to the table partitioning during the DMO, as described in section *Table Partitioning as of Start Release 7.00* of the DMO guide for SAP HANA, for the downtime-optimized conversion you have to consider the following additional aspects:

- As table content is already migrated during uptime, you can set a breakpoint between phases `EU_CLONE_MIG_OPTDMO_CREATE` and `EU_CLONE_MIG_OPTDMO_INI_RUN` in order to change the partitioning for the application tables directly on the SAP HANA database.
- You must not do any manual partitioning on tables VBFA or KONV. This is handled by SUM automatically.

3.7 Shadow Field Handling

The downtime-optimized conversion uses a technique called shadow field handling to save database reduce spacememory requirements and to improve performance.

Overview

The basic idea of this technique is to handle structural table changes by adding new or changed fields with a specific name pattern to the original table and hiding them by a database view that represents the original structure. These hidden fields are the so-called shadow fields. During the SUM downtime, the old fields

are dropped and the shadow fields renamed to their DDIC names. The shadow fields are filled by update statements for all existing rows in the table with the content from the original fields.

In nZDM, this happens in phase `RUN_IOC_TOUCH`. Rows changed via update or insert in the production system automatically update the shadow fields by using a database trigger on the table.

In downtime-optimized DMO, R3load transfers the data and replicating the changes. While doing this R3load also fills the shadow fields. No database triggers are needed.

How the shadow fields are filled from the original fields is either determined by the SUM for simple structural changes, for example a change from NUMC length 3 to length 5 only in nZDM, or by a table-specific application logic. The table-specific application logic is implemented in a so-called shadow field handler.

Phase `RUN_IOC_TOUCH` processes all tables in parallel. Each table is updated sequentially in packages with a default package size of 100.000. To avoid dead locks, the package size is automatically reduced in case a high change frequency on the original table is detected. The default package size can be changed by adding another value in `SAPup_add.par`, parameter `/shdfld/packagesize_touch`. If a table is partitioned, the update is executed in parallel per partition. Therefore, for a faster runtime, you can try to split large tables by increasing the number of partitions. The parallel execution depends on the number of configured ABAP processes. For more information about partitioning, see sub-section *Partitioning* under Planning.

The Main Difference of Shadow Field Handling between nZDM and downtime-optimized DMO Scenarios

In the nZDM scenario, the original table is changed to an artificial name and replaced by a database view with the original fields. This avoids copying the data to a new table and replaying all changes done on the original table. Instead, the database trigger fills with each change the shadow fields automatically. All other rows are handled by phase `RUN_IOC_TOUCH`.

In downtime-optimized DMO, the tables need to be copied to the new database, therefore, the shadow field handling is applied on the temporary system: The table is copied with data in its original structure. After that, the shadow fields are added and the touch is executed. A database view with the original fields is created. The data replication inserts data into this database view causing the database trigger to fire and, thereby, to fill the shadow fields.

In nZDM, pure structural changes are picked up for shadow field handling, if possible. In downtime-optimized DMO, these structural changes are handled by R3load more efficiently directly during initial load and replication of data.

There are several statically classified application tables in the downtime-optimized conversion that are always be handled with the shadow field technique, the most prominent are tables VBFA and KONV.

Special Tables KONV and VBFA

The conversion of table `VBFA` and `KONV` (in target release replaced by PRCD_ELEMENTS) is always performed with the use of the shadow field handling. As these tables can be very large in the systems, it is very important to define appropriate partitioning as described in this guide.

Checking Progress of Touch Update

The progress bar of the SUM UI provides details about the state of the running touch jobs.

3.8 Homogeneous Database Migration Option for SAP ERP on SAP HANA

For a source system on SAP HANA, the Software Update Manager 2.0 SPS 18 allows a homogeneous migration from SAP HANA to SAP HANA also for downtime-optimized conversion. We recommend that you use this migration instead of the in-place case (keeping the source SAP HANA database).

Using a seperate SAP HANA database as your target provides a better performance of the downtime-optimized conversion approach, as SUM uptime activities on the target SAP HANA database do not run on the same host as the end-user activity on the source system.

3.9 Software Update Manager Toolbox

The Software Update Manager Toolbox (SUM Toolbox) provides you with various tools and reports in order to support your activities in SUM.

The SUM Toolbox is realized as an ABAP-based transaction with the transaction code `SUMTOOLBOX` or `STBX`. It allows you to execute specific tools and reports for different software maintenance scenarios using the Software Update Manager, with a focus on the downtime-optimized approaches.

> ⓘ Note
>
> For a downtime-optimized conversion run, it is required that the SUM Toolbox is already implemented in the SAP ECC source system.
>
> Furthermore, you need to present the results of the impact analysis by providing the `ZDIMPANA.ZIP` file in your download directory folder. For more information, refer to SAP Note 2481983.

For more information about the SUM Toolbox and how to implement it, refer to SAP Note 3092738.

3.10 Specific Phases Having a Long Runtime or Resulting in a Timeout

During the uptime processing of a downtime-optimized conversion run, you may experience phases that have a long runtime or even result in a timeout. If you wonder why this happens, and whether it is safe to simply repeat the phase after such a timeout, this section provides technical background for this case.

For the uptime data conversion of the downtime-optimized conversion, the concept of Transactional Consistent Recording (TCR) ensures that a consistent set of database table content is converted, although the various table entries may have been migrated at different points in time. TCR is based on database triggers which must be set by SUM on all respective tables without time difference. The technology to achieve this depends on the following source database types:

- **For Source Database Types Other than SAP HANA or Oracle**
  SUM executes several lock table statements during and after database trigger creation in phase `SQLRUNTASK_RRC_TRIGGERS`. We recommend that you execute the trigger creation phase `SQLRUNTASK_RRC_TRIGGERS` and all other phases that run up to phase `RUN_DISABLE_TCR_HOOK` during a time where your system use is minimal. The less user activity is happening on the system, the faster those phases will be finished. This is especially true if you have tables that are changed very frequently and that are also part of the change recording with database triggers.
- **For Source Database Types SAP HANA or Oracle**
  SUM checks for open transactions without using lock table statements, so user activity between phases `SQLRUNTASK_RRC_TRIGGERS` and `RUN_DISABLE_TCR_HOOK` is not that critical. However, the less user activity is happening on the system, the faster those phases are completed. To set a timestamp; SUM uses the following phases:
  - SQLRUNTASK_RRC_TRIGGERS
  - SQLRUNTASK_RRC_LOCK_AFTER_TRIGGER_CRE
  - SQLRUNTASK_RRC_LOCK_AFTER_TRIGGER_ACT
  - RUN_RRC_TCR_STORE_COMMIT_ID
  - RUN_TNXMON_DOC_CREFR_SET
  - RUN_TNXMON_DOC_CREFR_CHECK
  - RUN_TNXMON_RRC_CRE_SET
  - RUN_TNXMON_RRC_CRE_CHECK
  - RUN_TNXMON_RRC_ACT_SET
  - RUN_TNXMON_RRC_ACT_CHECK
  - RUN_RRC_TCR_STORE_COMMIT_ID

  Afterwards, SUM monitors open transactions during these phases. It verifies that on all relevant tables no transactions are open that were created before the saved timestamp. The check phase waits until no such locks exist anymore. If this check phase takes more than one hour, the SUM stops with a timeout. For each timeout, the phase can be repeated, until finally no more open locks are found, and SUM processing continues. Every transaction started after the previously set timestamp is not relevant for the current wait phase.

4 Preparation

This part of the guide contains information about the preparations that you perform before starting the downtime-optimized conversion.

> → Recommendation
>
> You can prepare for the conversion while the system is in production operation. Careful preparation of the conversion procedure is the best guarantee for a successful run.

The Software Update Manager supports you by making a range of checks automatically in the preparation roadmap steps (*Initialization*, *Extraction*, *Configuration*, and *Checks*). To make sure that all requirements are met, run the preparation roadmap steps of Software Update Manager as soon as possible. The successful execution of the preparation roadmap steps is a prerequisite for starting the roadmap step `Preprocessing`.

You can reset and repeat the preparation roadmap steps as often as you need by choosing *Reset* before you start the actual procedure in roadmap step `Preprocessing`.

4.1 Relevant SAP Notes

For the downtime-optimized conversion, you require some additional information that is not included in this document.

This information is in a range of SAP Notes from the SAP Support Portal at http://support.sap.com/notes. Check these SAP Notes before you start with the preparations.

> ⚠ Caution
>
> Make sure that you always have the latest version of the SAP Notes.

SAP Note Number	Title
3619822	Executing and Monitoring Downtime-Optimized Conversion to SAP S/4HANA with SUM 2.0 SP 24
3620006	Database migration option (DMO) of SUM 2.0 SP 24
3572082	Central Note - Software Update Manager 2.0 SP 24

4.2 Preparation of Software Update Manager

Before starting the update with the downtime-optimized conversion to SAP S/4HANA, perform all the preparation steps mentioned in the SUM guide.

The Software Update Manager (SUM) controls the entire procedure, from checking the system requirements and importing the necessary programs through stopping production operation until production operation is resumed. The procedure is divided up into a number of different roadmap steps. The roadmap steps are in turn divided into phases. The successful completion of a phase is a precondition for the success of all subsequent phases.

For more information, see the *SUM guide* mentioned in Integration/Further Documentation [page 5]. See especially the following sections:

- *3 Introduction* > *The Software Update Manager*
- *6 Running the Software Update Manager* > *Working with the SUM Tool* > *Downloading and Unpacking the Software Update Manager*
- *6 Running the Software Update Manager* > *Working with the SUM Tool* > *Starting and Restarting the Software Update Manager*

4.3 Project Sequence for the Downtime-Optimized Conversion

Before you execute the downtime-optimized conversion procedure of your SAP ERP system, plan and prepare your project for the downtime-optimized conversion to SAP S/4HANA by performing several test runs.

Context

The first run is essential to create the Customizing and to determine additional tables to be migrated in uptime.

Procedure

1. Perform a regular conversion run **without** downtime-optimized conversion procedure.
    1. Copy your SAP ERP system to a sandbox system.
    2. Perform a regular conversion on the sandbox system without the downtime-optimized conversion procedure. Analyze which application tables have a long downtime migration time and can be moved to the uptime with the downtime-optimized conversion procedure.
    3. Define your Customizing settings and perform all SAP Finance-specific IMG steps on your new SAP S/4HANA sandbox system as described in SAP Note 2332030.

Your Customizing settings and IMG activities are recorded in transport requests that you need to include in your subsequent conversion runs as a customer buffer file.

> **Note**
>
> It is mandatory to use the **customer buffer** when performing a downtime-optimized conversion run. For more information, see SAP Note 1759080.

4. We recommend that you include all transport requests that are part of the SAP S/4HANA conversion project. This includes not only the SAP Finance Customizing but also any other Customizing, especially any retrofit changes, modification adjustments, and new custom development that is needed for the production go-live. We recommend that you avoid any changes that are intended to be released after the go-live. This ensures an optimized business downtime and system consistency at the same time.
   To ensure CTI buffer consistency, use the Customer Transport Integration Wizard. For more information, see SAP Note 1759080.

   > **Note**
   >
   > The integration of customer transports can lead to additional read-only restrictions in the production system. See section *Customer Transport Integration (CTI) for VDAT/CDAT with Zero Downtime Option (ZDO) for SAP S/4HANA and Downtime Optimized Conversion (DoC)* of SAP Note 1759080. Use the impact analysis to identify potential conflicts. For more information, see SAP Note 2402270.

   If a read-only restriction in production is not acceptable, perform the following mitigation: Exclude the table from uptime handling by adding the table to `DIR_PUT>/var/CRRTABLIST.LST`: `E <Tablename>`
   The following potential side effects are possible:
   - A table which is relevant for the uptime conversion cannot be excluded and leads to an error message in the subsequent SUM execution. In this case the read-only restriction in the production system cannot be avoided. To continue the SUM run, remove the table from the `CRRTABLIST.LST`, when the error eccurs.
   - If the uptime exclusion is successful, the table is not available for the delta Customizing activities on the temporary system. If you encounter an issue during the delta Customizing activities which results in a conflict, you need to get SAP Support. Open an incident on component BC-UPG-DTM-TLA.

2. Perform a test run **with** downtime-optimized conversion procedure.
   1. Copy your SAP ERP system to a test system.
   2. -live
   3. Perform one to several runs of downtime-optimized conversion on the test system.
   4. During the test run, create test data before the downtime, as mentioned in section Test Data for SAP S/4HANA Finance During the Test Run [page 32].
   5. Include your Customizing requests, which you generated on the sandbox system, in the downtime-optimized conversion procedure with the `Customer Transport Integration` feature of the Software Update Manager on your test system. This feature is described in the *SUM guide* in section *4 Planning* > *Integration of Customer Transports*.
   6. **Only relevant for source systems not yet on SAP HANA database:** Enter the path and name of the file containing the selected large tables that should additionally be migrated during uptime.

→ Recommendation

In order to refine your table selection, we recommend that you perform the test run with downtime-optimized conversion for several times.

3. Perform a productive run **with** downtime-optimized conversion procedure:

   Perform now an update with the downtime-optimized conversion procedure on your production system including your Customizing requests in the downtime-optimized conversion procedure with the *Customer Transport Integration* feature of SUM.

4.4 Prepare Your Financial Data for the Downtime-Optimized Conversion

Before starting your downtime-optimized system conversion, get your financial data for the daily processes in `Financials`, `Asset Accounting`, or `Controlling` ready.

Perform all required preparation steps as described in section *Prepare Phase: Getting Your System Ready for the Conversion* of the guide *Converting Your Accounting Components to SAP S/4HANA*. This guide is attached to SAP Note 2332030.

Follow all steps of subsection *Check and Reconcile Your Data* and all steps of subsection *Period-End Closing Activities* up to the section *Directly before the technical downtime, perform the following activities*.

4.5 Preparation of Database Migration Option

Before starting the update with the downtime-optimized conversion to SAP S/4HANA, perform all the preparation steps mentioned in the DMO guide.

→ Recommendation

Only relevant if SAP HANA is NOT the database of your source system:

See the *DMO guide* mentioned in Integration/Further Documentation [page 5] and follow here especially

- the preparation steps mentioned in section *Preparation*
- the preparation steps for the `Benchmarking Tool` mentioned in section *Migration Tools* > *Benchmark Migration*

4.6 Test Data for SAP S/4HANA Finance During the Test Run

During the `Preprocessing` roadmap step, shortly before the downtime begins, we recommend that the users of the financial department create test bookings.

These test bookings should be bookings for the daily processes in `Financials`, `Asset Accounting`, or `Controlling`. They are used as test data to test the conversion to SAP S/4HANA.

4.7 Moving Long-Running XCLA Processing to the Uptime

SUM phase `XPRAS_AIMMRG` executes XCLAs (and XPRAs) during the downtime of a system conversion run.

If you are using the downtime-optimized conversion aproach, you have the option to select specific XCLAs to be processed in uptime. For more information about how to select appropriate XCLAs for the uptime processing and how to configure SUM for this approach, see SAP Note 2778832.

4.8 Considerations for Long-Running Batch Jobs in Phase “RUN_WAIT_FOR_BATCHES”

If you want to avoid long waiting times in phase RUN_WAIT_FOR_BATCHES (although this happens during uptime processing), we recommend that you ensure that long-running batch jobs start after phase `RUN_ACTIVATE_TCR_HOOK`.

You have the following options:

- You plan the SUM execution to reach phase `RUN_ACTIVATE_TCR_HOOK` when no long-running jobs are running, for example by setting a breakpoint for that phase.
- You make sure that long-running jobs start after this phase, for example by scheduling those jobs for a start time after that phase execution.

All batch jobs that have started after phase `RUN_ACTIVATE_TCR_HOOK` are not relevant for phase `RUN_WAIT_FOR_BATCHES` and thus will not contribute to the wait time for that phase.

The technical reason is that phase `RUN_ACTIVATE_TCR_HOOK` sets the dynamic profile parameter dbsl_iuuc_end_transaction_hook to enable Transactional Consistent Replication (TCR). Dynamic profile parameter changes are not directly consumed by running work processes (such as batch jobs). They will be considered once the work process finishes.

5 Roadmap Steps with Downtime-Optimized Conversion Features

This part of the guide provides information about the update and conversion procedure using Software Update Manager.

It describes conversion-specific actions during the following affected roadmap steps:

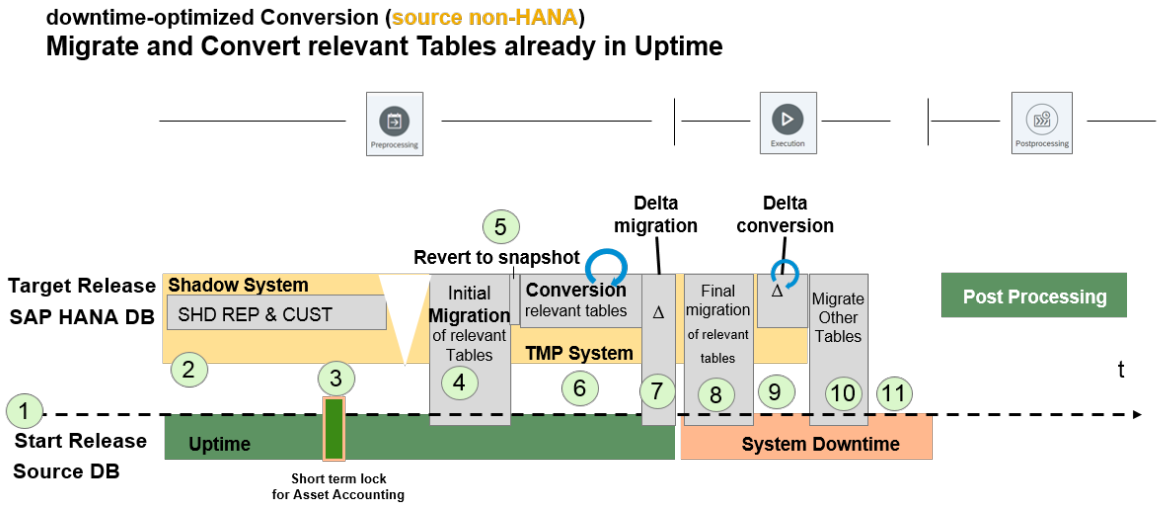
- Configuration [page 35]
- Preprocessing [page 35]
- Execution [page 36]

> **Note**
>
> You must have completed all required planning and preparation actions before you start the Software Update Manager.

In the following, we describe the main steps and phases of the downtime-optimized conversion procedure. The first section deals with the downtime-optimized conversion for source systems running on a database that is not SAP HANA.

The subsequent section regards the downtime-optimized conversion for source systems that run on an SAP HANA database.

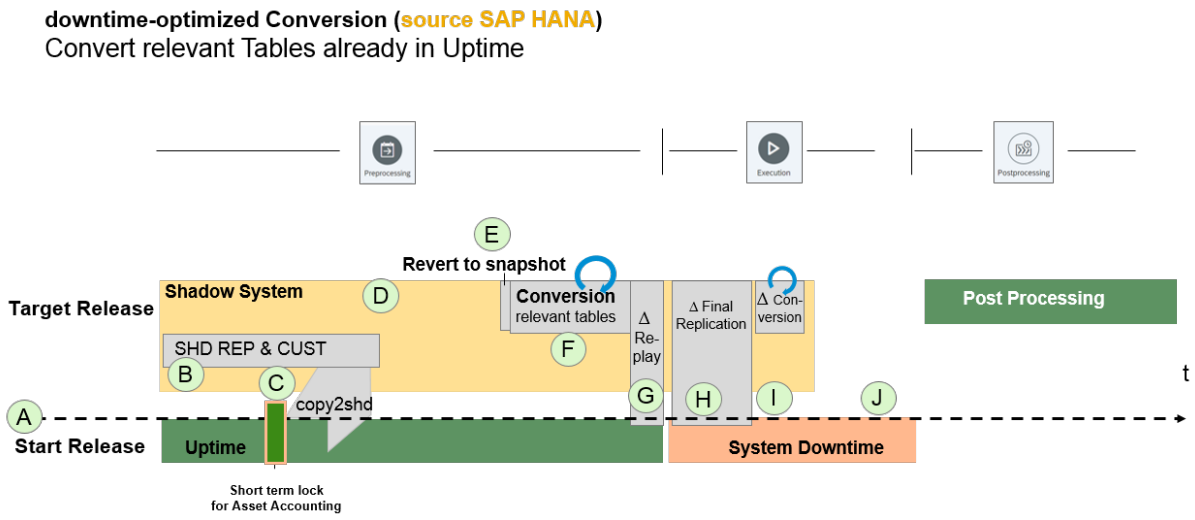
Downtime-Optimized Conversion for Source Systemss on non-HANA Database



1. You start the downtime-optimized conversion procedure in the Software Update Manager.
2. At the beginning of the `Preprocessing` roadmap step, the shadow repository is created.

3. Asset accounting lock to obtain a transactional consistent state of `Asset Accounting` before cloning.
4. Initial transfer of tables starts.
5. Revert transfer to revert snapshot triggers.
6. Conversion of relevant tables starts.
7. Delta migration of relevant tables.
8. In the `Execution` roadmap step, the remaining delta of relevant tables is migrated.
9. Conversion of remaining delta of relevant tables.
10. Migration of remaining tables.
11. Final activities of the downtime phase such as kernel switch and remaining update.

Downtime-Optimized Conversion for Source Systems on SAP HANA Database



A. You start the downtime-optimized conversion procedure in the Software Update Manager.

B. At the beginning of the `Preprocessing` roadmap step, the shadow repository is created.

C. `Asset Accounting` lock to obtain a transactional consistent state of `Asset Accounting` before cloning.

D. Copy tables to shadow.

E. Revert transfer to revert snapshot triggers.

F. Conversion of relevant tables starts.

G. Delta transfer of relevant tables.

H. In the `Execution` roadmap step, the final delta transfer is performed.

I. Conversion of remaining delta of relevant tables.

J. Final activities of the downtime phase such as kernel switch and remaining update

5.1 Configuration

In the *Configuration* roadmap step, the Software Update Manager asks for all information required to run the update with the downtime-optimized conversion.

The following actions are specific to the update with the downtime-optimized conversion:

- **Inclusion of Customizing Requests**
  To include your Customizing requests in the downtime-optimized conversion procedure, the Software Update Manager asks you for the path and name of your customer buffer file. The definition of clients for which table logging is turned on is retrieved from the parameter `RECCLIENT` of the transport profile file, or from system profile parameter `rec/client`. Depending on your requirements, you enter one of the following values for parameter `recclient`:

Value of Command Option	Meaning
RECCLIENT=off	No logging. It is the default setting.
RECCLIENT=all	Logging always occurs for all clients for client-specific tables.
RECCLIENT=<Client1>, <Client2>	For client-specific tables, logging occurs in the clients listed here.

5.2 Preprocessing

In the *Preprocessing* roadmap step, the Software Update Manager creates necessary data required for the uptime migration.

SUM creates

- a consistent copy of all tables that are read or written by the conversion
- all required Customizing, master data, and all application tables

The database triggers record the changes of all copied tables. SUM transfers all recorded changes to SAP HANA database. After the uptime migration has finished, SUM starts to replicate all recorded changes to SAP HANA until entering downtime.

The following actions are specific to the update with the downtime-optimized conversion:

- **Lock of `Asset Accounting` in Phase `MAIN_SHDRUN/REQ_SMIG_APPL_LOCK`**
  In this phase, leading tables of `Asset Accounting` are locked by database freeze trigger to prevent new postings. Confirm the dialog by choosing *Next*.
  During the `Asset Accounting` lock, perform the following steps for the daily processes in `Asset Accounting`:
  1. Perform closing for periodic asset postings using report `RAPERB2000` or transaction `ASKB`.
  2. Calculate the plan values for depreciation using transaction `AFAR` in the productive system. This means you can avoid RC errors during the migration.

This `Asset Accounting` lock ends in phase `MAIN_SHDRUN/REQ_SMIG_APPL_UNLOCK`. Confirm the dialog by choosing *Next*.

- **Finance Customizing**
  To support additional changes that were made after the inclusion of your Customizing request, proceed as follows :
  1. Log on to the application client of your update system with your business user. This can be the shadow instance, if your source database is already on SAP HANA, or the temporary instance in case of any other source database.
     Make sure to apply the changes in all necessary clients. Before doing that, unlock your update system.
  2. To perform the following activities in the update system, call transaction `UPG_SFIN_CUST`:
     - Migrate Material Ledger Customizing
     - Check Assignment of Material Ledger Types to Valuation Areas

  > ⓘ Note
  >
  > Note that the Customizing in the shadow or temporary system using theses activities is only an exception and not a regular activity.

5.3 Execution

In the *Execution* roadmap step, the Software Update Manager records and replays all remaining changes.

Afterwards, the Software Update Manager starts the delta conversion for all records that have not been part of uptime conversion.

Perform all remaining preparation steps for the daily processes in `Financials`, `Asset Accounting`, or `Controlling` as described in section *Prepare Phase: Getting Your System Ready for the Conversion* of the guide *Converting Your Accounting Components to SAP S/4HANA*. This guide is attached to SAP Note 2332030.

Follow the steps of chapter *Period-End Closing Activities* starting from section *Directly before the technical downtime, perform the following activities*, and all steps of the subsequent chapter *Document Your Posting Data*.

6 Follow-Up Activities

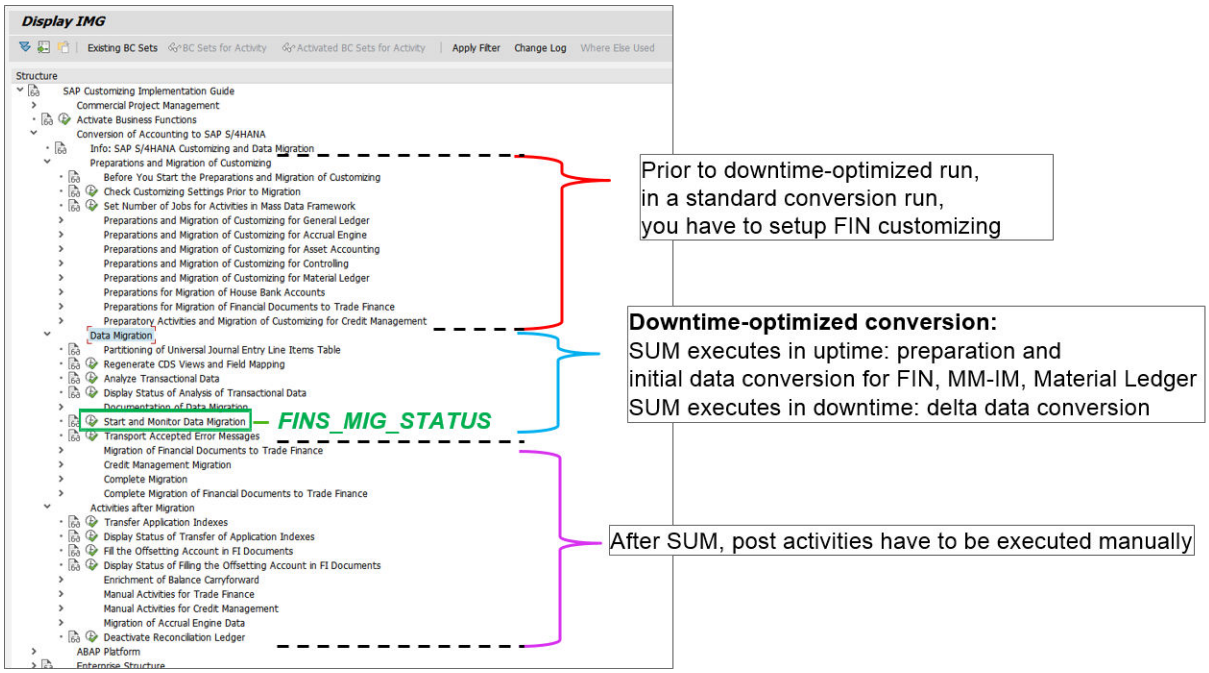
This part of the guide contains information about the follow-up activities that you need to perform after you have completed the update and the conversion procedure.

6.1 Performing IMG Steps After the Conversion

After having completed the downtime-optimized conversion procedure, perform all Implementation Guide steps that are listed under *Conversion of Accounting to SAP S/4HANA* > *Data Migration* > *Start and Monitor Data Migration*.

Proceed as follows:

1. Call transaction SPRO in your SAP system.
2. In the SAP Customizing Implementation Guide under *Conversion of Accounting to SAP S/4HANA* > *Data Migration* > *Start and Monitor Data Migration*, perform all IMG steps that are listed here and summarized with the lowest curly bracket in lilac color in the figure below:



6.2 Performing Further Follow-Up Activities

Afterward completing the required IMG steps, perform all necessary follow-up activities that are described in the associated *SUM guide* and *DMO guide*, which are mentioned in Integration/Further Documentation [page 5]

7 Troubleshooting

This section contains information about how to proceed when you want to correct known problems that have occurred during the downtime-optimized conversion procedure.

7.1 Generic Troubleshooting Activities

This part of the document contains additional information about all generic activities that you have to do in order to peform a successful troubleshooting.

7.1.1 Finding the Relevant Log Files

When an issue occurs, we recommend that you view the SUM UI as your very first step.

In many cases the SUM UI displays an excerpt of a log file that is a good starting point for analyzing the errors.

For more log files, navigate to `<DIR_PUT>/abap/log`, either using transaction `AL11` in your ABAP system or using your preferred tool on OS level (for example WinSCP or Putty) and sort the files in a descending order by time (for example on UNIX using the command `ls -ltr`).

In general, check first the most recent files, since the SUM updates the log files frequently. In some cases, it also makes sense to check the most recent log files in `<DIR_PUT>/abap/tmp`.

7.1.2 Analyzing Short Dumps

To find out the root-cause of errors in the SAP system, the analysis of short dumps is a good starting point.

You can use the tools of the ABAP dump analysis (transaction ST22) to list the ABAP runtime errors that have occurred in an ABAP system.

If a short dump exists in the relevant time frame of the issue analyzed and the root cause is not clear or if you cannot fix it on your own, create an incident on the application component listed in the header of the short dump in transaction ST22. If no application component is listed, you can use component `BC-UPG-DTM-TLA` to ask for help from SAP.

7.1.3 Checking the Job Overview

The transaction SM37 allows you to monitor the jobs running in the SAP system. However, it is also useful to find information about aborted jobs.

To check for aborted jobs in your system, on the *Simple Job Selection* screen, make the following entries:

1. Under *Job Name*, enter `*`.
2. Under *User Name*, enter either `DDIC` or `*`.
3. Under *Job Status*, select: *Canceled*.
4. Under *Job Start Condition*, enter a relevant timeframe (for example current date in "From" and "To" to get all aborted jobs of the current day).

7.1.4 Checking the Database Logs

Specific downtime-optimized conversion activities (for example FIN conversion, MM-IM conversion) are executed in the shadow/temporary system during the uptime steps and in the downtime prior to phase `EU_SWITCH_MIG` and in the upgrade target system after phase `EU_SWITCH_MIG`.

Regardless of scenario, either it is anyDB or an in-place conversion, in case of issues that are caused by the database, you can find valuable information written in diagnostic files of the SAP HANA database.

To access the files, as a prerequisite, you need <SID>adm credentials to check logs at OS level or you need TRACE ADMIN or DATA ADMIN to view or delete log files in the HANA Studio.

At OS level these diagnosis files (trace and log files) are generated at the following location of each node/ hostname: `/usr/sap/<SID>/HDB<nn>/<hostname>/trace/DB_<TENANT_ID>`.

You can also execute the shell command alias `cdtrace` to access these files.

In case of unusual situations, you can start the analysis with `indexserver*` and `indexserver_alert*` logs.

For more information about diagnostic files, see section *Diagnosis Files* of the HANA Administration Guide on the SAP Help Portal.

7.1.5 Checking the DBA Cockpit

Transaction DBACOCKPIT provides various functions to utilize during upgrades.

It provides an overview of used memory and the user can check object properties for tables and indexes, or whether such objects are missing.

SQL queries can be run manually (via the SQL Editor) and the user can check running database processes.

If the customer is suffering from long running steps in SUM, the SQL history can be collected from the DBACOCKPIT to analyze for tweaking (Expensive Statements).

The diagnosis files also give clues. They provide results of consistency checks, executed statements, loads, and more.

7.1.6 Oracle Only: Deprecated Datatypes

If there are tables in the customer system that contain fields of the deprecated data type `LONG`, these tables could lead to errors during a downtime-optimized conversion.

The SUM creates a snapshot table with the new data type `BLOB`. However, since the database triggers created by SUM write directly to the snapshot table, this leads to an error because in the initial table the field is still of data type `LONG`. This issue currently only exists on Oracle databases.

For tables relevant for downtime-optimized conversion, the customer must convert the data types of these tables from `LONG` to `BLOB` prior to starting the downtime-optimized conversion. If a data type conversion is not possible on the customer system, create an incident on component `BC-UPG-TLS-TLA` for a detailed analysis.

7.1.7 Handling Function Call Failure

Mostly, the reason for a function call failure lies in issues in the infrastructure.

If you can rule out that infrastructure is the culprit and all evidence points at the working of the respective function itself and you are only left with creating a ticket, collect the source code and version of the function. Proceed as follows:

1. Call transaction **SE37**.
2. Choose *Source* for the source code.
3. Choose *Utilities* > *Versions* > *Versions Management*.
4. To speed up analysis, provide this information to the processor of your incident.

7.2 Troubleshooting for the Finance and Material Management-Inventory Management Migration

The Finance migration and the Material Management-Inventory Management (MM-IM) migration are delta-enabled steps and are executed in the shadow or temporary system in 2 steps:

1. Full, during uptime
2. Delta, during downtime

For details and the most frequent error messages, see SAP Note 2714344 *Financial data migration to SAP S/4HANA: Most frequent Error Messages – Information and Recommendations*.

7.2.1 Finance Migration

If an issue occurs, the errors are visible in the shadow or temporary system on the productive client. The best starting point for analysis is transaction `FINS_MIG_STATUS`.

We highly recommend that you always analyze these errors with the assistance of a Finance application expert.

Issues can occur due to the following reasons:

- Missing Customizing:
  - The actual migration process cannot start without a Customizing and is highly dependent on data constellation.
  - The Customizing transport does not fit anymore due to recent changes in the source system's configuration. The safest approach in this case is to reset the SUM and provide an appropriate Customizing.
- Inconsistent data:
  - There might be situations when manual intervention in data or wrong archiving left the data in the source system in an inconsistent state from the data model's perspective.
  - Depending on the situation, the Finance expert could accept the errors and, in the next cycle, correct the data in the source system. Also in this case, a safe approach is to reset the SUM and perform data correction in the source system, if time allows.

Depending on the migration step in the Finance migration process, you can create an incident on components `FIN-MIG`, `FIN-MIG-ML` or `FIN-MIG-AA`.

For more information about prerequisites and restrictions for Finance, see section *Downtime and Business Restrictions* of this guide.

7.2.2 MM-IM

If issues occur during the MM-IM conversion, this is reported in the SUM log file `LONGPOST.LOG` with different error messages.

Once the SUM process is completed, you can identify messages such as the following:

- NSDM_MIG_CONTROL 082 MKPF group verification with unprocessed groups, for more information, see SAP Note 2538948.
- NSDM_MIG_CONTROL 078 Verification might be incomplete: MKPF processing is not finished.
- NSDM_MIG_CONTROL 090 Aggregate group verification with unprocessed groups, for more information, see SAP Note 2538948.
- NSDM_MIG_CONTROL 075 Verification might be incomplete: aggregate processing is not finished.

See SAP Note 2538948 *Verification (SUM) fails with post-processing error* and execute report `NSDM_CONV_STATISTICS` after the processing steps of the conversion have been finished to extract detailed information.

Analyze migration issues that arise during phases `../SUBMOD_SMIG_UT_MMIM/PARRUN_SMIG_UT_MKPF1_S4`, `../SUBMOD_SMIG_UT_MMIM/PARRUN_SMIG_UT_MKPF2_S4` or `../SUBMOD_SMIG_UT_MMIM/RUN_SMIG_UT_MKPF3_S4` by viewing logfiles `NSDM_MIG_MKPF_XXX.<SID>`.

Depending on the error, you can open an incident on component `MM-IM-GF-MIG` or other `MM-IM-*` components.

For more information about prerequisites and restrictions for MM-IM, see SAP Note 2281657.

7.3 Performance Impact

From the performance impact perspective, the HANA in-place scenario is the most critical.

The HANA in-place scenario is the most critical because uptime activities are being executed on the same database in parallel with production operations.

Particularly in this scenario, we highly recommend that you perform the following activities:

- Choose the number of DIA/BTC parallel processes in the SUM configuration steps in such a way that out-of-memory or admission control situations (CPU & memory) on the database side that cause disruption on the running business are avoided.
- See SAP Note 1969799 *SQL Statement Collection for SAP HANA* that provides several SQL statements and helps you monitor the usage of the database and identify potential issues.
- Actively monitor the system with standard monitoring tools, such as the following:
  - ABAP: Transactions in the shadow system: SM66, ST02, ST22, SM04
  - DB: HANA Studio, HANA Cockpit
  - Operating system: top, nmon, glance

7.4 Monitoring the Replication

SUM uses two techniques for replication of data: ABAP batch processes (running on the shadow/temporary system) and R3load processes.

The replication is started in SUM phase `PROCESS_REPLICATE_RRC_START` and you can monitor its status in the SUM Process Control Center. To do so, proceed as following::

1. Open the *Process Control Center* area of SUM Utilities in the SUM UI.
2. In order to see what replication processes are currently active in the respective sections ABAP Replication Monitor and R3load Replication Monitor, view section *CRR Control Center*.
3. To stop or to start the replication, you can use the buttons on the right side. Next to the *Stop* button, there is also a log file button that you can use to directly open the corresponding replication log file.

The CRR Control Center can also be used to change the number of batch processes and R3load processes. However, the number of batch processes is limited by the SUM process parameters.

If you want to add additional batch processes, when the replication is progressing too slowly, follow the steps described in the following sub-section *Reconfiguring the Temporary System to Add Resources* of this guide.

7.5 Correcting Errors in Specific Phases

This sub-section decsribes errors encountered in various phases of the downtime-optimized conversion and how to solve them.

7.5.1 Handling Missing Partitioning in Phase MAIN_MIGOPT/SUBMOD_SMIG_PREP_DMO/JOB_FINS_UPG_MIG_PREPARE

Context

In phase `MAIN_MIGOPT/SUBMOD_SMIG_PREP_DMO/JOB_FINS_UPG_MIG_PREPARE`, you may encounter the following error messages with recommendations to partition Finance tables:

- A4 EFINS_UTIL 016 Read accesses successfully redirected from "FAP_V_CMP_HBACC" to "V_FAP_CMP_HBACC"
- A2EEFINS_FI_MIG 706 Expected ACDOCA size is "4,8" to "5,1" billion records; partitioning needed

Procedure

Proceed as described in SAP Note 2289491.

7.5.2 Handling Errors in Phase MAIN_MIGOPT/SUBMOD_SMIG_PREP_DMO/JOB_FINS_UPG_MIG_PREPARE

Context

In phase `MAIN_MIGOPT/SUBMOD_SMIG_PREP_DMO/JOB_FINS_UPG_MIG_PREPARE`, the CDS views cannot be created due to failed authority check for user DDIC. You receive the following error messages:

- 20181017 102302 You don't have sufficient authorization for this activity
- 20181017 102302 Job canceled after system exception ERROR_MESSAGE

To solve the errors, proceed as follows:

Procedure

1. Log on to the temporary system.
2. Execute report **RSUPG_WRP_REGEN_SAPALL**.
3. Repeat the phase.

7.5.3 Handling Errors in Phase SMIG_SFIN_MIGPREP_TRC_EVA

Context

In uptime phase `SMIG_SFIN_MIGPREP_TRC_EVA`, you may encounter errors with the following error messages:

- A2EESMIG 327 Illegal write access on table `WBCROSSGT`
- A2EESMIG 327 Illegal write access on table `ANEK`
- A2EESMIG 327 Illegal write access on table `ANKB`
- A2EESMIG 326 Illegal read access on table `T005SEPA`
- A2EESMIG 327 Illegal write access on table `T082AVIEWB`
- A2EESMIG 327 Illegal write access on table `T093`
- A2EESMIG 327 Illegal write access on table `T093B`
- A2EESMIG 327 Illegal write access on table `T093T`
- A2EESMIG 327 Illegal write access on table `T095`
- A2EESMIG 327 Illegal write access on table `T095B`

Procedure

Create an incident on component `BC-UPG-DTM-TLA`. The SAP development team will check if the downtime-optimzed procedure metadata for the uptime conversion can be updated and if the errors can be ignored.

7.5.4 Handling Errors in Phase REQ_PARTITIONING_MATDOC

Context

In phase `REQ_PARTITIONING_MATDOC`, shortly before downtime you are prompted to partition table `MATDOC`. In case of downtime-optimized conversion, the table `MATDOC` already contains data, partitioning requires time as all the data has to be moved by the SAP HANA DB into the defined partitions.

Procedure

1. When table `MATDOC` is still empty and to avoid long runtimes at this point of the conversion, create the partitioning already beforehand.
2. To do so, follow the instructions of SAP Note 2259038.

7.5.5 Handling Errors in Phase PARRUN_SMIG_SFIN

Numerous errors may occur during the Preprocessing roadmap step in phase `MAIN_SMIG_UT_MIGRATION/SUBMOD_SMIG_SCENARIOS/SUBMOD_SMIG_SFIN/PARRUN_SMIG_SFIN`

Context

You can find a summary of these errors in the upgrade log.

To view detailed error messages, proceed as follows:

Procedure

1. Log on to the affected client of the temporary system in SAPGui with user DDIC while the temporary system is locked.
2. Start transaction FINS_MIG_STATUS to view the detailed error log.

   The use of various transactions is limited in the temporary system as not all application tables are available. You might not be able to fix missing customizing using IMG activities, for example. This needs to be fixed in the customizing source system and included in the customizing transport for the customer buffer for the next iteration.
3. Decide accordingly what activity you need to perform:

- Use of transaction FINS_MIG_STATUS to analyze detailed error messages from data migration and to accept these errors.
- Use of transactions SE16 or SE16H for root cause analysis.
- Use of transaction FINS_MIG_STATUS to repeat migration steps after corrections.
- Once you have analyzed and fixed the errors, lock the temporary system in case it was unlocked and restart the upgrade and the conversion procedure.

7.5.5.1 Handling Errors During Uptime Finance Conversion in Phase PARRUN_SMIG_SFIN [AAA Phase)

Context

The following errors are encountered during the uptime Finance conversion in the phase PARUN_SMIG_SFIN:

- Chart of depreciation 9MCA: Assign a target ledger group to area 41
- Chart of depreciation 9MUK: Assign a target ledger group to area 41

You can see these error messages from the Finance migration workbench using transaction **`FINS_MIG_STATUS`**.

Procedure

1. To avoid this error, clean up the inconsistencies prior to starting the SUM run.
2. If the phase already failed, a possible workaround is to delete the target ledger group 41 and 51 from the chart of depreciation 9MCA and 9MCUK using function module `AA_AFABER_DELETE` from temporary system. However, it must be confirmed by the customer's Finance team that the affected target ledger groups are not used.

7.5.5.2 Handling Shortdump in Phase PARRUN_SMIG_SFIN

Context

During phase `PARRUN_SMIG_SFIN`, you may face the `TSV_TNEW_PAGE_ALLOC_FAILED` dump.

Procedure

1. To solve this error, follow the instructions as described in SAP Note 2532269.

2. If this doesn't help increase parameters `em/initial_size_MB, abap/heap_area_total` and `abap/heap_area_nondia` to higher values for the shadow or temporary system and repeat the failed phase.

7.5.5.3 Handling Performance Issues in Phase PARRUN_SMIG_SFIN

Context

Due to small table buffer size for the temporary system, almost all database selects are ignoring the table buffer and execute a query on the database.

Procedure

1. Stop the temporary system.
2. Open the instance profile of the temporary system on OS-level: `<DIR_PUT>/system/<SID>/SYSTMP/profile/<InstanceProfile>`
3. Delete the following parameters:
   - zcsa/table_buffer_area
   - zcsa/presentation_buffer_area
   - zcsa/db_max_buftab
   - rtbb/buffer_length
   - rtbb/max_tables

   Removing the parameters will enable the default setting for the temporary system.
4. Start the temporary system.
5. Repeat the phase.

7.5.6 Handling Short Dumps in Phase PARRUN_SMIG_UT_MKPF1_S4

Short dumps in phase `PARRUN_SMIG_UT_MKPF1_S4` can occur with error message `TSV_TNEW_PAGE_ALLOC_FAILED`.

Context

Memory consumption can reach more than 1GB per batch process during phase `PARRUN_SMIG_UT_MKPF1_S4`.

Procedure

1. Make sure that the ABAP heap memory has at least 1 GB for each available batch process or the number of batch processes needs be reduced.
2. Use report **RSMEMORY** to set the ABAP heap memory parameters abap/heap_area_total, abap/heap_area_dia and abap/heap_area_nondia to an appropriate higher value.

7.5.7 Handling ST22 Dump due to Freeze Trigger

Context

Customer scenario: When HR is adding a new hire to SAP and proceeds to add the banking record, the Bank Key (Transit #) entered is validated against table BNKA.

The Bank Key must exist on table BNKA or else it cannot be saved to the HR record. Occasionally field HR hits this scenario and advises HC.

Procedure

1. With the help of the impact analysis, outline this issue to the customer as a restriction for the productive run.
2. Make sure that the customer enforces organizational governance to adhere to this restriction as soon as the hard freeze starts.

7.5.8 Handling Errors During Uptime Replication

Context

You have observed that R3load buckets are aborted during the uptime replication monitoring.

Procedure

1. During the phase MIGRATE_OPTDMO_REPLAY_RUN, check for the error directly in the R3load bucket log.
2. Reschedule the failed buckets via the corresponding buttons in the CRR Control Center.

7.5.9 Handling DBSL Error 99 in Phase SCEXEC_GRANT_RSV_DT

Context

When a DBSL error 99 occurs in phase `SCEXEC_GRANT_RSV_DT`, proceed as follows:

Procedure

To reduce the number of parallel SQL processes for phase `SCEXEC_GRANT`, set the parameter `/SCEXEC_GRANT/parprocs/DDL = 1 in DIR_PUT/abap/bin/SAPup_add.par`.

7.5.10 Handling Errors in Phase MAIN_NEWBAS/PARMVNT_UPG

Context

In phase `MAIN_NEWBAS/PARMVNT_UPG`, the following objects cannot be created, as they already exist:

- SQL-error *322-cannot use duplicate view name: EHPRCV_ANL_**
- SQL-error *329-cannot use duplicate name of function or procedure: REPM_SALOR*

Procedure

1. Drop view `EHPRCV_ANL_*` manually on the SAP HANA database.
2. Drop function `REPM_SALORD` manually on the SAP HANA database.
3. Repeat the phase.

7.5.11 Handling Missing Database Objects in Phase MAIN_POSTPROC/RUN_RSDB02CK_END_DMO

Context

In phase MAIN_POSTPROC/RUN_RSDB02CK_END_DMO the following objects might be missing:

- 2EETG002 View "/GRC/V_BANK_ACCT" does not exist in the database

- 2EETG002 View "COPC_V_T012_SKB1" does not exist in the database
- 2EETG002 View "COPC_V_T035D" does not exist in the database

Procedure

Activate these objects using transaction SE11.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon  : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

www.sap.com/contactsap

© 2025 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see https://www.sap.com/about/legal/trademark.html for additional trademark information and notices.

THE BEST RUN SAP