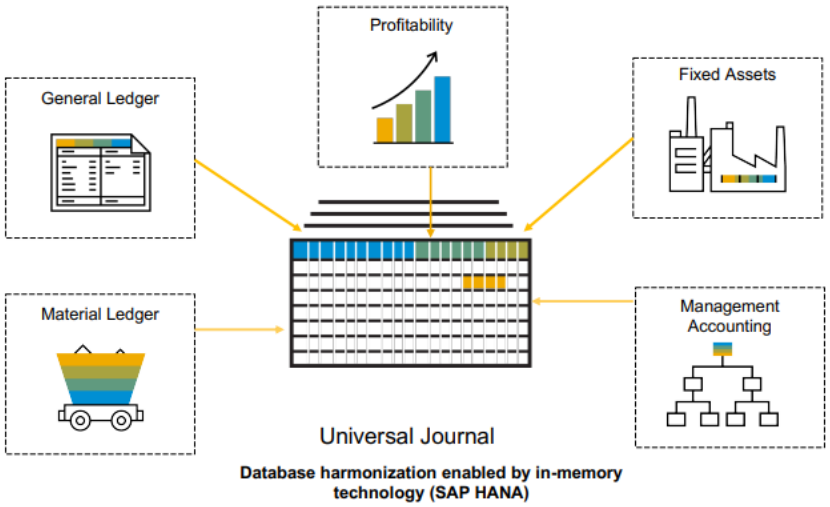
1. **S/4HANA Finance - Overview**
2. **SAP S/4HANA JVA On-Premise Edition - Roadmap**
3. **S/4HANA JVA On-Premise Edition 1909 – Innovations – JVA on ACDOCA Solution**
4. **SAP S/4HANA JVA-Central Finance Integration - Overview**

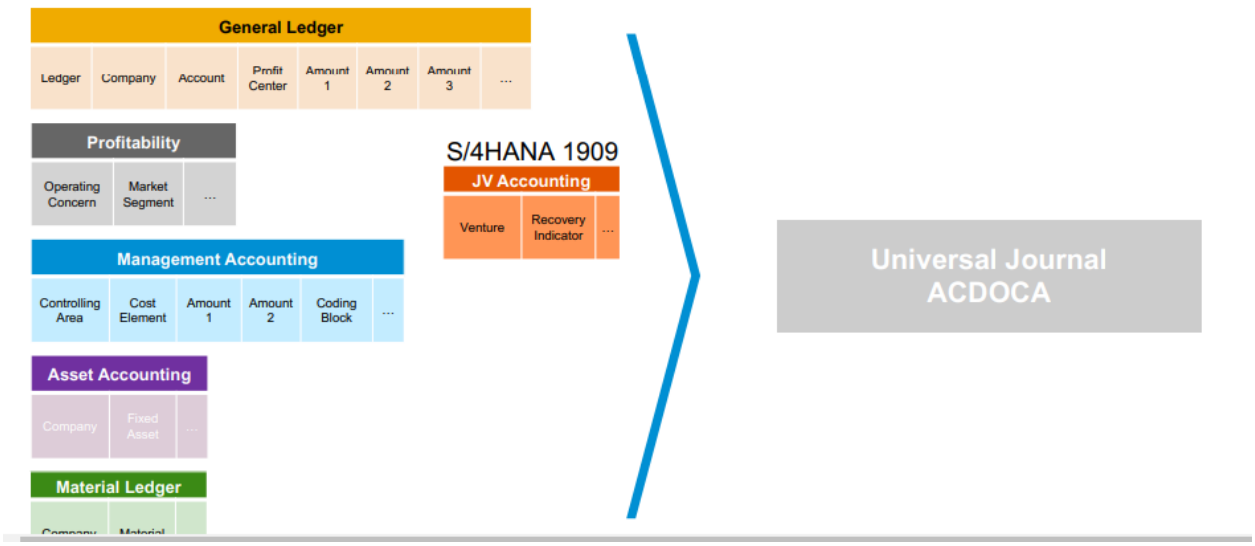
S/4HANA Finance

S/4HANA Finance On-Premise Edition

Business Architecture: Fundamental simplification in the architecture of accounting



S/4HANA Finance On-Premise Edition
S/4HANA: Universal Journal



S/4HANA JVA On-Premise Edition 1909 – JVA on ACDOCA *
Universal Journal Single Line Item Table ACDOCA with JVA Fields

Adding JVA Fields in ACDOCA to replace the JVA Special Ledger tables

Universal Journal – JVA Fields													
Venture	EG	RI	Prod Month	Billing Month	Operations Month	Partner Venture	Partner Equity Group	Cutback Run ID	JVIND	Sender RI	Orig Account	Orig Cost Object	....

Billing Indicator is a field in ACDOCA now

New Data Model Adaption

- Actual (i.e. Cash Call, JV Funding)
- Month End (i.e. Cutback, Equity Change, Equity Adjustment, Farm In/Out)
- Billing (Invoice and Expenditure Detail Extract and Hardcopy Billing)
- Reporting (i.e. CDS View based JVA line item reporting Fiori app)

Joint Venture Accounting
Key Business Processes

Operator's View

Venture Sourcing
- Accounts Payable
- GR/IR
- Open Item & Clearing

Venture Funding
- Accounts Receivable
- Treasury & Banking
- Project Budgeting
- AFE Process

Master Data Management
- JVA MD
- FI, CO, PS MD

Accounting Approach Configuration
- Billing Basis
- Operator as Partner
- Operator Oriented

1. Venture Cost Capturing → 2. Pre-Cutback Activities and Cutback - Allocation of Venture Costs → 3. Venture Partner Billing

Actual Processes
- Activity Allocation
- Asset / Material Transfers
- Cash Calls
- Non-Operator Invoice Processing

Period Processes
- FXD (RXD, UXD)
- Cost Allocation
  - Assessment
  - Distribution
  - Settlement
- Overhead
- JV Funding
- Suspense
- Equity Adjustment
- Cutback
- CI/NPI Processing
- Netting
- Billing (Invoice, Expenditure Detail, SOE and Billing Hardcopy)
- Government Reporting - PSC (Royalty, Recoverable Costs, Profit, Income Tax)

Finance Month End Process (FI /CO and JVA)

Cost Transfers → Foreign Currency Valuation → Cost Allocations → Overhead Calculations (PCO) → Equity Adjustments → Cutback → Settlement AuC Own Share → Partner Netting → Final Settlement → Billing → Corporate Close

SAP S/4HANA JVA On-Premise edition – JVA on ACDOCA
Product road map overview – Key innovations

Lab Preview

1809 – Recent innovations[1]
- UI Visual Harmonization (JVA Fiori VH tiles)

1909 – Recent innovations[1]
- Move to new JVA data model based on Universal Journal
- JV Funding
  - New Venture Bank Switching Function
  - Interest Calculation
- Improved JVA process handling
  - Cutback – Invoice per partner posting
  - Cutback –Reversal
  - Cash Call
  - Farm In & Farm Out
  - Equity Change
  - Suspense MD Display
- Fiori Venture MD object page app
- Fiori JOA MD object page app

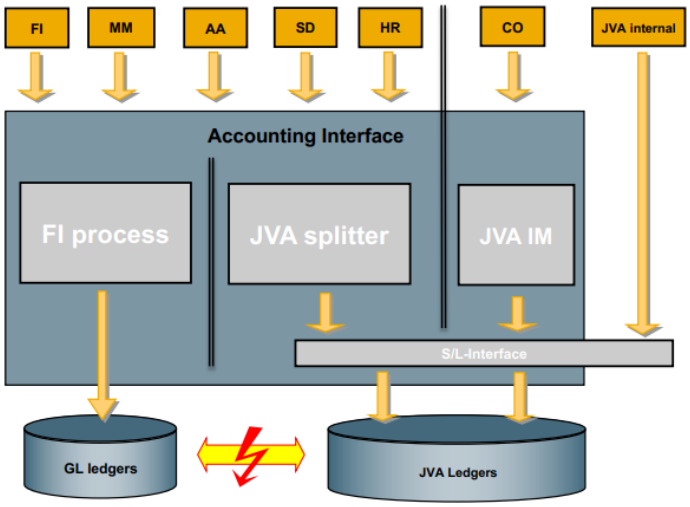
2020 – Product Direction[1,2]
- JVA on ACDOCA Migration support (FPS1)
- JVA PCO Overhead enhancement
- JVA-cFIN Integration (FPS1)

2021 – and Future Releases[1,2]
- PSA Rollup Rule ACDOCA adaption
- JVA Finance Overview Fiori app
- Plan Data Processing (first version)

S/4HANA JVA On-Premise edition 1809 and prior releases (e.g. ECC6 etc..)
Classic JVA data workflow in ERP based on Special Ledgers without Finance integration

Document Split in JVA
(Integration Manager)

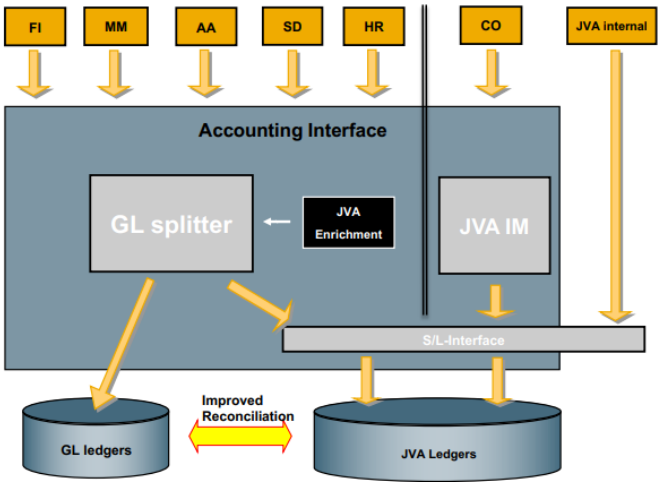
2 sources of true for reporting
- Finance
- JVA



S/4HANA JVA On-Premise edition 1809 and prior releases (e.g. ECC6 EhP6 etc..)

JVA data in ERP with NewGL-JVA integration based on Special Ledger data model

Document Split in FI-GL
2 sources of true for reporting
- Finance
- JVA



Switchframework:
- JVA BF
- JVA-NewGL BF

Improvement:
- FI document for JVA internal documents
- GJ90 for RXD is replaced by CO-FI reconciliation
- GJ91 for UXD and GJNO for B/S valuation are replaced by FAGL_FC_VAL

SAP S/4HANA JVA On-Premise Edition 1909 - JVA on ACDOCA

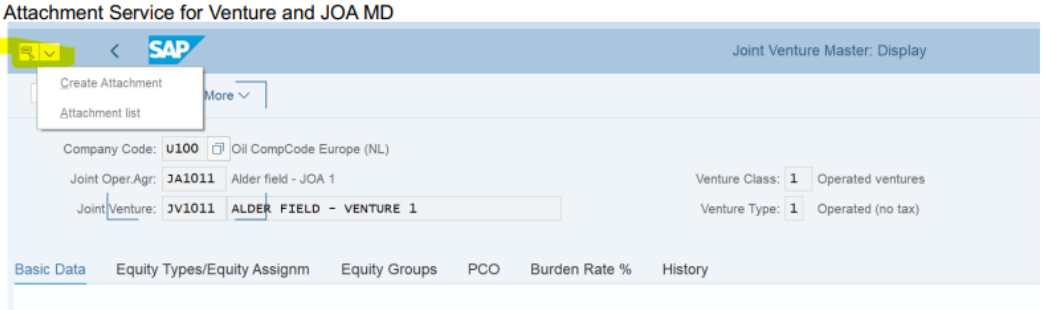
Changed JVA Processes

- JVA Special Ledger fields are replaced by the Universal Journal ACDOCA table
  - Thus the JVA Reporting and Partner billing are done out of the ACDOCA table (except for JVA plan data).
  - 100% reconciliation between CO, FI and JVA
- **JVA Document Split** (JVA Integration Manager)
  - is replaced by Finance document split (Venture and EG are becoming a mandatory split characteristic)
  - thus the venture document is replaced by finance document with split line item by Venture and EG
- **JVA internal postings** are updating Finance and generates Finance documents
- **FXD**
  - RXD replaced by FI-CO integration and Venture document split (by Venture and EG) in Finance
  - UXD and B/S Item Valuation replaced by standard Finance UXD program
- **JVA Bank Switching**
  - replaced by JV Funding
  - enhanced by Interest calculation for interventure liabilities

SAP S/4HANA JVA On-Premise Edition 1909 – JVA on ACDOCA

Simplification & Innovation for Joint Venture Accounting on Universal Journal*

- Attachment Service for Venture and JOA MD



- Migration to S4HANA with Joint Venture Accounting running on ACDOCA

Version	7	**Type**	SAP Note
Language	English	**Master Language**	English
Priority	Recommendations / Additional Info	**Category**	Consulting
Release Status	Released for Customer	**Released On**	10.11.2021
Component	CA-JVA-JVA-IF (Integration Functionality)		

Please find the original document at **https://launchpad.support.sap.com/#/notes/ 2941622**

Symptom

Migration to S/4HANA with Joint Venture Accounting running on ACDOCA

Other Terms

JVA, Joint Venture Accounting

Reason and Prerequisites

As of the on-premise S/4HANA release OP1909, the Joint Venture Accounting can fully run on ACDOCA when the "JVA_ON_ACDOCA" business function is active. This is a radical change in the JVA data model, away from the separate JV ledgers to an ACDOCA-integrated data model. This note provides guidance to customers who run ERP or S/4HANA systems with the old JVA data model and answers the question if and how an existing productive system can be upgraded or migrated to S/4HANA with JVA running on ACDOCA.

Solution

Availability of the Solution

The JVA-integrated S/4HANA migration will be fully available as of release OP2022.

For the OP releases OP2020 and OP2021 the solution will be provided via the following notes:

3050563 - JVA on ACDOCA activation: Report to update the CBRUNID value in the ACDOCA table

3058813 - JVA on ACDOCA activation: Reports to update the ACDOCA table with internal JVA documents and with accounting data from the JVA

3082514 - Integration of JVA into the S4HANA migration

Notes 3058813 and 3082514 still have the pilot status and are released to individual customers on request only. So, if you cannot access these notes via transaction SNOTE, send a request to SAP via support ticket.

JVA on ACDOCA

The new JVA data model is called "JVA on ACDOCA". It is based on the Universal Journal (datase table ACDOCA), rather than on the JVA ledgers which were the basis of the classic JVA solution. The following note provides basic information on how a JVA on ACDOCA system needs to be configured. Before any migration to JVA on ACDOCA is planned, this note should be studied:

Types of migrations/terminology

In the SAP context, the concept of migration or conversion refers to a situation where a new software release based on a new data model is installed in an existing productive system with transactional data. Due to the new data model, the affected data need to be converted so that they are available in the way required by the new software release. This notes will provide you with information about the following migrations types:

1. Internal migration/conversion:
   A new software released with a new data model is installed in an existing productive system with transactional data. Before the upgraded software can be productive, the old data have to be converted to the new data model.
2. Switch scenario:
   The business function JVA on ACDOCA is switched on in an existing S4HANA system that is productive and has transactional data both in the ACDOCA and the JVA ledgers already.
3. Cross-system migration:
   A new software release is installed in a separate new system and the data in the old system(s) are transferred to the new system. SAP offers two solutions for this type of migration:
   - Central Finance: Both the historic and the new data in the source systems are transferred to the target system and reposted based on the new data model.
   - DMLT service: The historic data in the old system are copied to the new system. Then the copied data are converted to the new data model.
   This note will not handle these cross-system migrations. If you need guidance for this type of migration, please contact the SAP support for Central Finance or DMLT.

General Remarks on Migration Scenarios

Any ERP system that is upgraded to S/4HANA needs to undergo a technical data conversion: Its main task is to collect and convert the data that are saved in separate tables in the ERP system and to combine the data into the Universal Journal Entry table called ACDOCA. This data conversion is called "S/4HANA Finance Migration" or "SFIN conversion" (for more details and for links to instructions, see note 2294486). It is important to understand that this internal migration is a purely technical data conversion. It is not possible to switch on business feature at the same time, especially, it is not possible to switch on the GL splitter.

This has consequences for JVA customers because the JVA_ON_ACDOCA business functions requires an active GL splitter that is integrated with JVA. This again means that any ERP system that has not yet been activated for NewGL with an active GL splitter and with NewGL-JVA integration has to be migrated to this configuration first. In other words, for ERP customers that run the classic FI with classic JVA, there is no single step migration to S/4HANA with JVA running on ACDOCA! Before the system can be upgraded to S/4HANA, the NewGL migration has to be executed. It is important to understand that this step is by far the bigger and more demanding step.

Customers that are running an ECC system without NewGL-JVA integration and would like to avoid a two-step migration can alternatively set up a completely new system ("greenfield implementation") and do a cross-system migration by importing the historic transactional data into the new system. This can also be achieved through a DMLT service or through a Central Finance system environment. Both approaches have pros and cons.

There is no general answer to the question which approach is the best and which one goes with minimal risk and effort. The best migration path depends on the individual accounting requirements and needs to be figured out by the best internal accounting experts together with experienced(!) JVA and FI/CO consultants.

General Remarks on Switch Scenarios

When an existing S4HANA system already runs productively with classic JVA and when there is the plan to activate JVA on ACDOCA in this system, we are calling this switch scenario. In this case, the main challenge is to adapt the configuration according to the requirements of JVA on ACDOCA and to deal with the disruption between the transaction data posted before and after the the switch to JVA on ACDOCA was made.

Description of Scenarios

The tables below list the different migration and switch scenarios and provide information if a scenario will be supported by SAP standard tools and if there are restrictions or issues to be expected (see the disclaimers below each table). As the target system, an up-to-date S/4HANA system is assumed (as of release OP2020) where the GL splitter is active and JVA is running on ACDOCA.

Migration Scenarios

The following table contains scenarios where the source system is an ECC system that is supposed to be upgraded to S4HANA.

Note: For instructions how to figure out if the NewGL-JVA integration is active in your system, see annex 1 below.

Source Release/System	Remarks/Details
(1) ERP Classic GL + Classic JVA	Supported (after pre-migration to NewGL). Migration Steps: (1) Ensure the consistency of the data in the classic JVA database tables (see annex 2 below). (2) In the given ECC system, upgrade to NewGL with GL splitter and JVA-integration. For this task standard migration tools are available. Important note: This is a big task requiring a lot of time and effort! (3) Upgrade the ERP system to S/4HANA. (4) Implement notes 3050563, 3058813 and 3082514. (5) Switch on the JVA on ACDOCA business function. (6) Configure the system according to the instructions given in note 3044558. (7) Execute the S/4HANA migration. (8) Run special JVA reports to update the ACDOCA table with JVA data. The details are described in notes 3050563 amd 3058813. (9) All self-developed customer reports and programs need to be adapted to the new data model. If, for technical or legal reasons, the data in the ACDOCA table cannot be perfectly updated with JVA data (step 7), all programs and reports (including the self-developed ones) need to use the approprate switch views. (For more details, see note 3082514.) (10) Execute full regression testing for all accounting processes. See the disclaimer for migration scenarios below!
(2) ERP NewGL with GL splitter + Non-integrated classic JVA	Supported with severe restrictions! There is no guarantee that the results of the GL splitter can be combined with the results of the JVA splitter. (However, a solution tailored to the individual situation might be available via SAP SLO support.) A disruption of the migrated data and the data posted after the migration is to be expected. It is therefore very likley that the usage of compatibility or switch views will be required. Migration Steps: (1) Ensure the consistency of the data in the classic JVA database tables (see annex 2 below). (2) Upgrade the ERP system to S/4HANA. (3) Implement notes 3050563, 3058813 and 3082514. (4) Switch on the JVA on ACDOCA business function. Via transaction GJACTACD, define the start date as of when JVA on ACDOCA should be active for new postings (for more details, see annex 4 below). (5) Configure the system according to the instructions given in note 3044558. (6) Execute the S/4HANA migration (without JVA integration!). (7) Run special JVA reports to update the ACDOCA table with JVA data. The details are described in notes 3050563 amd 3058813. (8) All programs and reports (including the self-developed customer programs and reports) need to

	use the approprate switch views. (For more details, see note 3082514.) (9) Execute full regression testing for all accounting processes. See the disclaimer for migration scenarios below!
(3) ERP NewGL with GL splitter + NewGL-JVA integration	Supported. Migration Steps: (1) Ensure the consistency of the data in the classic JVA database tables (see annex 2 below). (2) Upgrade the ERP system to S/4HANA. (3) Implement notes 3050563, 3058813 and 3082514. (4) Switch on the JVA on ACDOCA business function. (5) Configure the system according to the instructions given in note 3044558. (6) Execute the S/4HANA migration. (7) Run special JVA reports to update the ACDOCA table with JVA data. The details are described in notes 3050563 amd 3058813. (8) All self-developed customer reports and programs need to be adapted to the new data model. If, for technical or legal reasons, the data in the ACDOCA table cannot be perfectly updated with JVA data (step 6), all programs and reports (including the self-developed ones) need to use the approprate switch views. (For more details, see annex 5 below.) (9) Execute full regression testing for all accounting processes. See the disclaimer for migration scenarios below!

Disclaimer for Migration Scenarios

No guarantees can be given that the migration will produce data in the ACDOCA table that perfectly reflect the data in the JV ledgers. A special and critical problem can be caused through JVA substitutions for the JV ledger. If cost objects are changed through JV substitutions and if this affects cost elements or secondary cost elements, the cost object in the JV tables will not be take over to ACDOCA table in the single items to ensure consistency with the classic CO data.

Due to the complexity of the issue, experienced JVA consultants/experts should be involved in the planning and execution of a migration project.

Switch Scenarios

The following table contains scenarios where a productive system is already running on S4HANA.

System Configuration	Remarks/Details
(4) S/4HANA with GL splitter + NewGL-JVA integration + cost objects defined as split criteria	Supported with restrictions (see the disclaimer below)! Migrations steps: (1) Switch on the JVA_ON_ACDOCA business function. Via transaction GJACTACD, define the start date as of when JVA on ACDOCA should be active for new postings (for more details, see annex 4 below). (2) Configure the system according to the instructions given in note 3044558. (3) Run special JVA reports to update the ACDOCA table with JVA data. The details are described in notes 3050563 amd 3058813. (4) All self-developed customer reports and programs need to be adapted to the new data model. For legal reasons (see below), it is to be expected that the data in the ACDOCA table cannot be perfectly updated with JVA data (step 4), all programs and reports (including the self-developed ones) need to use the approprate switch views. (For more details, see annex 5 below.) (5) Execute full regression testing for all accounting processes that are affected by the new JVA data model, that is, you need to run tests for all JVA processes and for all processes that create FI or CO postings.
(5) S/4HANA with GL splitter + NewGL-JVA integration + cost objects not defined as split criteria	Supported with restrictions (see the disclaimer below)! Migrations steps: (1) Switch on the JVA_ON_ACDOCA business function. Via transaction GJACTACD, define the start date as of when JVA on ACDOCA should be active for new postings (for more details, see annex 4 below). (2) Configure the system according to the instructions given in note 3044558. (3) Run special JVA reports to update the ACDOCA table with JVA data. The details are

	described in notes 3050563 amd 3058813. (4) All self-developed customer reports and programs need to be adapted to the new data model. For legal reasons (see below), it is to be expected that the data in the ACDOCA table cannot be perfectly updated with JVA data (step 4), all programs and reports (including the self-developed ones) need to use the approprate switch views. (For more details, see annex 5 below.) (5) Execute full regression testing for all accounting processes that are affected by the new JVA data model, that is, you need to run tests for all JVA processes and for all processes that create FI or CO postings.
(6) S/4HANA without GL splitter + Classic JVA (Assumption: S/4HANA migration has already taken place, without switching on the GL splitter beforehand)	Not supported! Reason: While the GL splitter can be switched on in S/4HANA, no support for JVA is available for this functionality.
(7) S/4HANA Central Finance (JVA-enabled) with GL splitter + NewGL-JVA integration + cost objects defined as split criteria	Supported with restrictions (see the disclaimer below)! Migration steps: (1) Switch on the JVA_ON_ACDOCA business function. Via transaction GJACTACD, define the start date as of when JVA on ACDOCA should be active for new postings (for more details, see annex 4 below). (2) Configure the system according to the instructions given in note 3044558. (3) Run special JVA reports to update the ACDOCA table with JVA data. The details are described in notes 3050563 amd 3058813. (4) All self-developed customer reports and programs need to be adapted to the new data model. For legal reasons (see below), it is to be expected that the data in the ACDOCA table cannot be perfectly updated with JVA data (step 3), all programs and reports (including the self-developed ones) need to use the approprate switch views. (For more details, see annex 5 below.) (5) Execute full regression testing for all accounting processes that are affected by the new JVA data model, that is, you need to run tests for all JVA processes and for all processes that create FI or CO postings.
(8) S/4HANA Central Finance + JVA not active	Not supported! Reason: Switching on JVA in a cFin system will disrupt the consistency between the historic data and the new data. Additionally, there will be issues in clearings and reversals.

Disclaimer for Switch Scenarios

To switch on JVA on ACDOCA in an S4HANA system that already has productive data in the ACDOCA table will cause some kind of disruption, esp. because of missing cost object on balance sheet accounts in GL and different split results in GL and JVA! There is no guarantee that processes that retrieve data from periods or fiscal years before the switch date will run correctly in all cases. In especially, inception-to-date processes will be affected. The success of the switch project and the degree of disruption depend on the specific situation in the given system. Therefore, no gurantee can be given by SAP that the switch is possible or will fulfill all customer requirements.

A special and critical problem can be caused through JVA substitutions for the JV ledger. If cost objects are changed through JV substitutions and if this effects cost elements, it might not be possible to take over the cost object in the JV tables to ACDOCA because the cost object posted to CO must be retained. At the very least, the customer must decide between the cost object in CO or the one in the JV ledgers. It will not be possible to have both cost objects.

Even though special switch views will be available that combine the JVA ledger data before the switch with the GL/ACDOCA data posted after the switch (step 5), there is no guarantee that these views will cover all cases of disruption.

Due to the complexity of the issue, experienced JVA consultants/experts should be involved in the planning and execution of a switch project.

ANNEX

(1) Migration scenarios: How to determine if the NewGL-JVA integration is active?

The NewGL-JVA integration is active in your your ECC system when the following criteria are met:

(a) The business function JVA_GL_INTEGRATION is active. This can be checked via transaction SFW5.

(b) Run transaction GJGL: All JVA company codes are listed; and the checkbox "NewGLsplit" is ticked for the each company code.

(c) The venture (VNAME) is defined as a split criteria in the GL splitter settings.

(2) How to ensure data consistency in the classic JVA database tables?

(a) Run transaction GCGS (report RGUSLSEP) for ledgers 4A and 4C and for all JVA company codes and all fiscal years to check if the sum totals in table JVTO1 match the single item amounts in table JVSO1. The program should not find any issues.

(b) However, if transaction GCGS detects any issues, the data in the JVTO1 table need to be fixed by running two SL reports:

1. Delete the inconsistent summary lines by running transaction GCDE for the relevant ledgers, years, and company codes.
Important remarks:
- You MUST run GCDE for complete years!
- In the frame "Deletion scope" tick only the first checkbox "Total records"! The other three options MUST NOT be ticked - otherwise you will delete the single line items in JVSO1 or JVPO1 !

Note: If transaction GCDE can not be executed for ledger 4c, 4d, or 4f, run the report RGUDEL00 via transaction SE38, instead. If this does not work either, execute the generation report RGZZGLUX via transaction SE38 to create the programs needed.

2. Recreate the summary lines by running report RGUREP01 (via transaction SE38) for the relevant ledgers, years, and company codes.

(3) Migration scenarios: How to activate the JVA integration into the S4HANA migration/SFIN conversion?

This step is relevant for migration scenarios only, whereas it is not relevant for switch scenarios.

By default, the integration of JVA with the S4HANA migration (also called "SFIN conversion") is **off**. In special cases this might be the correct setting! So, please check together with the most experienced inhouse JVA experts and external consultants if the JVA integration is required. If, for example, the data in the JV tables cannot be trusted (which might be true in special cases), it could be appropriate to not activate the JVA integration.

To actually activate the JVA integration, execute the following steps:

(a) Launch transaction SM30 for Maintenance View 'V_JVA_OPTIONS' in change mode.

(b) To switch **on** the JVA integration into the S4HANA migration, add the following entry:

Company Code	Option Name	Option Value
*	INTEGRATION_IN_S4HANA_MIGRATION_ACTIVE	X

(4) Switch scenarios: How to set a start date for JVA on ACDOCA?

To maintain the start date as of when JVA should fully run on ACDOCA, launch transaction GJACTACD (or, if the transaction code is not available, launch transaction SM30 for maintenance view JVAONACDOCAACTIV) and enter an entry like the following one (sample):

Activity ID	Date	Fiscal Year	Period
START	01.01.2023	2023	001

The above setting is important tells the system as of which posting date/fiscal period the processing functionality should use the logic for JVA_ON_ACDOCA and post to the ACDOCA table only. Moreover, the setting is used by the switch views that are explained explained below: As of the defined start date the switch views will take the data from the ACDOCA table, while they will take all data that were posted before the start date from the classic JV tables.

(6) If you need to run the classic JVA transaction for the old data model even after JVA on ACDOCA has got active for postings in the new period, that is, if you have to run the classic JVA month-end transactions for a previous period were the classic JVA was still active, you can make an additional setting that is used to determine if the classic transaction or the JVA-on-ACDOCA transaction is launched.

To maintain the start date for the JVA-on-ACDOCA transactions, launch transaction GJACTACD (or, if the transaction code is not available, launch transaction SM30 for maintenance view JVAONACDOCAACTIV) and enter an entry like the following one (sample):

Activity ID	Date	Fiscal Year	Period
START_TRANSACTIONS	01.02.2023	2023	002

To maintain a different start date for the JVA-on-ACDOCA transactions for specific users, launch transaction GJACTACD (or, if the transaction code is not available, launch transaction SM30 for maintenance view JVAONACDOCAACTIV) and enter an entry like the following one (sample):

Activity ID	Date	Fiscal Year	Period
START_TRANSACTIONS_[user id]	15.02.2023	2023	002

(5) Instructions for the usage of the CDS views shipped with note 3082514

Note provides 3082514 quite a few CDS views (including compatibility views) that can or need to be used for specific purposes when JVA is run on ACDOCA. The main views and its usages are explained in the tables below.

Remark on the views with the suffix "_SWITCH": The so-called switch views combine the data in the ACDOCA table with the data in the corresponding classic JVA table (JVSO1, JVSO2, JVTO1, JVTO2). The start date for JVA on ACDOCA stored in database table JVONACDOCAACTIVE is used as the separation date: The accounting data with a posting date before the start date are retreived from the classic JVA table

whereas the data with a posting date as of the start date are taken from the ACDOCA table.

Use a switch view only when this is really required. Due to their nature, switch views have a lower performance than non-switch views that retrieve the data from one table only.

Important remarks for developers: When using SELECT commands for CDS views, do not retrieve all columns via "SELECT * " (unless this can really not be avoided). Instead, do always specify the columns to be retrieved from the database. The list of of columns should only include the data that are required in the follow-up processes. Through restricting the retrieved data this way, you will increase the performances of the data selects. As the HANA database is not line item-based but column-based, the performance of data selects are negatively correlated with the number of columns retrieved. Or, in other words, the selects will run the faster, the fewer columns are retrieved. This is especially true for CDS views because many views are based on a stack of other CDS views, and some of them even have their own, temporary persistance.

The provided CDS views *must not be modified* as they are partly used by standard migration processes! If you need different views for your special purposes, you can copy the basis standard view and create a new one in the customer name space.

Views that retrieve data from the ACDOCA table with the JVSO1/JVTO1 structure

CDS View Name / ABAP SQL Name	Data Content	Usage
JV_JVSO1_ACDOCA/ CJVSO1_ACD	View on the ACDOCA table that returns the retrieved data with the JVSO1 data structure. The amounts for ledger 4C are stored in the columns HSL_4C and KSL_4C.	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVSO1 and need to be quickly adapted to JVA on ACDOCA. This view should not be used in a switch scenario or in a migration scenario if it is not possible to update the ACDOCA table with all JVA-relevant data. For such scenarios use JV_JVSO1_ACDOCA_SWITCH (see below).
JV_JVSO1_ACDOCA_4A_4C/ CJVSO1_ACD_4A_4C	View on the ACDOCA table that returns the retrieved data with the JVSO1 data structure, separated by JV ledger. (The performance of this view is lower than the one of JV_JVSO1_ACDOCA.)	See above.
JV_JVSO1_ACDOCA_SWITCH/ CJVSO1_ACD_SW	Switch view that combines data from the ACDOCA table with data from the JVSO1 table. It returns the retrieved data with the JVSO1 data structure. The amounts for ledger 4C are stored in the columns HSL_4C and KSL_4C.	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVSO1 and need to be adapted to JVA on ACDOCA. The view should be used if the historic data in the ACDOCA table

		(historic data in migrations scenarios or data in closed fiscal years/periods in switch scenarios) cannot be completely updated with the correct JVA accounting data.
JV_JVSO1_ACDOCA_4A_4C_SWITCH/ CJVSO1_ACD4A4CSW	Switch view that combines data from the ACDOCA table with data from the JVSO1 table. It returns the retrieved data with the JVSO1 data structure, separated by JV ledger. (The performance of this view is lower than the one of JV_JVSO1_ACDOCA_SWITCH.)	See above.
JV_JVTO1_ACDOCA/ CJVTO1_ACD	View on the ACDOCA table that returns the retrieved data with the JVTO1 data structure. The amounts for ledger 4C are stored in the columns with the suffix "_4C".	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVTO1 and need to be quickly adapted to JVA on ACDOCA. This view should not be used in a switch scenario or in a migration scenario if it is not possible to update the ACDOCA table with all JVA-relevant data. For such scenarios use JV_JVTO1_ACDOCA_SWITCH (see below).
JV_JVTO1_ACDOCA_SWITCH/ CJVTO1_ACD_SW	Switch view that combines data from the ACDOCA table with aggregated data from the JVSO1 table (not from JVTO1!). It returns the retrieved data with the JVTO1 data structure. The amounts for ledger 4C are stored in the columns with the suffix "_4C".	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVSO1 and need to be adapted to JVA on ACDOCA. The view should be used if the historic data in the ACDOCA table (historic data in migrations scenarios or data in closed fiscal years/periods in switch scenarios) cannot be completely updated with the correct JVA accounting data.
JV_JVTO1_ACDOCA_4A_4C_SWITCH/ CJVTO1_ACD4ACSW	Switch view that combines data from the ACDOCA table with aggregated data from the JVSO1 table (not from JVTO1!). It returns the retrieved data with the JVTO1 data structure,	See above.

	separated by JV ledger. (The performance of this view is lower than the one of JV_JVTO1_ACDOCA_SWITCH.)	
JV_JVTO1_ACDOCA_SWITCH_2/ CJVTO1_ACD_SW2	Switch view that combines data from the ACDOCA table with aggregated data from the JVTO1 table. It returns the retrieved data with the JVTO1 data structure. The amounts for ledger 4C are stored in the columns with the suffix "_4C". Important restrictions: This view can only be used when the data in the JVTO1 table are consistent with the data in the JVSO1 table. Moreover, this view can only be used when the start date for JVA on ACDOCA corresponds to the first date of the fiscal year of the leading ledger (which is the case when the fiscal year is identical with the calender year and when January 1st is set as start date).	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVTO1 and need to be adapted to JVA on ACDOCA. The view should be used if the historic data in the ACDOCA table (historic data in migrations scenarios or data in closed fiscal years/periods in switch scenarios) cannot be completely updated with the correct JVA accounting data.

Views that retrieve data from the ACDOCA table with the JVSO2/JVTO2 structure

CDS View Name / ABAP SQL Name	Data Content	Usage
JV_JVSO2_ACDOCA/ CJVSO2_ACD	View on the ACDOCA table that returns the retrieved data with the JVSO2 data structure. The amounts for ledger 4D are stored in the columns HSL_4D and KSL_4D.	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVSO2 and need to be quickly adapted to JVA on ACDOCA. This view should not be used in a switch scenario or in a migration scenario if it is not possible to update the ACDOCA table with all JVA-relevant data. For such scenarios use JV_JVSO2_ACDOCA_SWITCH (see below).
JV_JVSO2_ACDOCA_4B_4D/ CJVSO2_ACD_4B_4D	View on the ACDOCA table that returns the retrieved data with the JVSO2 data structure, separated by JV ledger. (The performance of this view is lower than the one of JV_JVSO2_ACDOCA.)	See above.

JV_JVSO2_ACDOCA_SWITCH/ CJVSO2_ACD_SW	Switch view that combines data from the ACDOCA table with data from the JVSO2 table. It returns the retrieved data with the JVSO2 data structure. The amounts for ledger 4D are stored in the columns HSL_4D and KSL_4D.	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVSO2 and need to be adapted to JVA on ACDOCA. The view should be used if the historic data in the ACDOCA table (historic data in migrations scenarios or data in closed fiscal years/periods in switch scenarios) cannot be completely updated with the correct JVA accounting data.
JV_JVSO2_ACDOCA_4B_4D_SWITCH/ CJVSO2_ACD4B4DSW	Switch view that combines data from the ACDOCA table with data from the JVSO2 table. It returns the retrieved data with the JVSO2 data structure, separated by JV ledger. (The performance of this view is lower than the one of JV_JVSO2_ACDOCA_SWITCH.)	See above.
JV_JVTO2_ACDOCA/ CJVTO2_ACD	View on the ACDOCA table that returns the retrieved data with the JVTO2 data structure. The amounts for ledger 4D are stored in the columns with the suffix "_4D".	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVTO2 and need to be quickly adapted to JVA on ACDOCA. This view should not be used in a switch scenario or in a migration scenario if it is not possible to update the ACDOCA table with all JVA-relevant data. For such scenarios use JV_JVTO2_ACDOCA_SWITCH (see below).
JV_JVTO2_ACDOCA_SWITCH/ CJVTO2_ACD_SW	Switch view that combines data from the ACDOCA table with aggregated data from the JVSO2 table (not from JVTO2!). It returns the retrieved data with the JVTO2 data structure. The amounts for ledger 4D are stored in the columns with the suffix "_4D".	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVSO2 and need to be adapted to JVA on ACDOCA. The view should be used if the historic data in the ACDOCA table (historic data in migrations scenarios or data in closed fiscal years/periods in switch scenarios) cannot be completely updated with the

		correct JVA accounting data.
JV_JVTO2_ACDOCA_4B_4D_SWITCH/ CJVTO2_ACD4BCSW	Switch view that combines data from the ACDOCA table with aggregated data from the JVSO2 table (not from JVTO2!). It returns the retrieved data with the JVTO2 data structure, separated by JV ledger. (The performance of this view is lower than the one of JV_JVTO2_ACDOCA_SWITCH.)	See above.
JV_JVTO2_ACDOCA_SWITCH_2/ CJVTO2_ACD_SW2	Switch view that combines data from the ACDOCA table with aggregated data from the JVTO2 table. It returns the retrieved data with the JVTO1 data structure. The amounts for ledger 4C are stored in the columns with the suffix "_4C". Important restrictions: This view can only be used when the data in the JVTO2 table are consistent with the data in the JVSO2 table. Moreover, this view can only be used when the start date for JVA on ACDOCA corresponds to the first date of the fiscal year of the leading ledger (which is the case when the fiscal year is identical with the calender year and when January 1st is set as start date).	Use this view for self-developed reports and views that were developed for classic JVA, are based on the data in table JVTO2 and need to be adapted to JVA on ACDOCA. The view should be used if the historic data in the ACDOCA table (historic data in migrations scenarios or data in closed fiscal years/periods in switch scenarios) cannot be completely updated with the correct JVA accounting data.

Views that return data from table JVSO1 with the ACDOCA structure

CDS View Name / ABAP SQL Name	Data Content	Usage
JV_ACDOCA_JVSO1/ CACDOCA_JVSO1	View on the JVSO1 table that returns the retrieved data with the ACDOCA data structure. The amounts of the JV ledger are stored in the corresponding ACDOCA fields with the same currency type.	Use this view for self-developed reports and views that need to retrieve data from the historic fiscal periods for special purposes. There will be hardly any use cases for this view. This view should anyway not be used in a switch scenario or in a migration scenario where data from ACDOCA und JVSO1 need to be combined. For such scenarios use JV_ACDOCA_JVSO1_SWITCH (see below).
JV_ACDOCA_JVSO1_SWITCH/ CACDOCA_JVSO1_SW	Switch view that combines data from the ACDOCA table with	Use this view for self-developed reports and views that retrieve data from the ACDOCA table and need to combine these with historic

	data from the JVSO1 table. It returns the retrieved data with the ACDOCA data structure. The amounts of the JV ledger are stored in the corresponding ACDOCA fields with the same currency type.	data from the JVSO1 table. This view is especially relevant if the historic data in the ACDOCA table cannot be completely updated with the correct JVA accounting data.

Views that return data from table JVSO2 with the ACDOCA structure

CDS View Name / ABAP SQL Name	Data Content	Usage
JV_ACDOCA_JVSO2/ CACDOCA_JVSO2	View on the JVSO2 table that returns the retrieved data with the ACDOCA data structure. The amounts of the JV ledger are stored in the corresponding ACDOCA fields with the same currency type.	Use this view for self-developed reports and views that need to retrieve data from the historic fiscal periods for special purposes. There will be hardly any use cases for this view. This view should anyway not be used in a switch scenario or in a migration scenario where data from ACDOCA und JVSO2 need to be combined. For such scenarios use JV_ACDOCA_JVSO2_SWITCH (see below).
JV_ACDOCA_JVSO2_SWITCH/ CACDOCA_JVSO2_SW	Switch view that combines data from the ACDOCA table with data from the JVSO2 table. It returns the retrieved data with the ACDOCA data structure. The amounts of the JV ledger are stored in the corresponding ACDOCA fields with the same currency type.	Use this view for self-developed reports and views that retrieve data from the ACDOCA table and need to combine these with historic data from the JVSO1 table. This view is especially relevant if the historic data in the ACDOCA table cannot be completely updated with the correct JVA accounting data.

Views that return data from the ACDOCA table for JVA purposes

CDS View Name / ABAP SQL Name	Data Content	Usage
JV_ACDOCA/ CJVACDOCA	View on the ACDOCA table with a subset of fields that are important for the cutback-related and equity group-	Use this view for self-developed reports and views that are completely based on the values in the ACDOCA table and that do not need to process any historic data that were created before JVA_ON_ACDOCA was switched on (greenfield

	related JVA processes	implementations; or processes retrieving values from the current or previous period only)
JV_ACDOCA_BILLING/ CJVACDOCA_BILL	View on the ACDOCA table with a subset of fields that are important for the billing-related JVA processes	Use this view for self-developed reports and views that are completely based on the values in the ACDOCA table and that do not need to process any historic data that were created before JVA_ON_ACDOCA was switched on (greenfield implementations; or processes retrieving values from the current or previous period only).

Software Components

Software Component	Release
SAP_FIN	618 - 618
SAP_FIN	619 - 619
SAP_FIN	720 - 720
SAP_FIN	730 - 730
EA-FIN	617 - 617
EA-FIN	700 - 700
S4CORE	100 - 100
S4CORE	101 - 101
S4CORE	102 - 102
S4CORE	103 - 103
S4CORE	104 - 104
S4CORE	105 - 105
S4CORE	106 - 106
EA-APPL	606 - 606
EA-APPL	616 - 616

This document is referenced by

Document by Irshad Rather