What's New in

Temenos Transact

December 2020

Table of Contents

Release Highlights

Analytics

Transact Data Hub » Spark Optimisation, Metrics Collection and Operational Monitoring

The Spark optimization feature in TDH allows users to perform data loading even with minimal infrastructure and upgrade on provisioning of additional resources. This feature is implemented:

- To use single landing process to process both Online and ILP events configured in DES.
- To use single ODS process to process both Online and ILP events of all applications configured in ODS.
- To use single SDS process to process both Online and ILP events of all applications configured in SDS.
- To upgrade or downgrade resource allocation for running ODS or SDS job based on requirement.
- To remove the start and stop functionality from ODS and SDS but the same functionality is handled internally during enabling or disabling applications.

In TDH, spark jobs handle each data storage process to populate data from source to target platforms. In this process, the spark metrics library helps to register each application process, push the metrics into an external database, and project it in Grafana dashboard. Spark can collect the variety of metrics by embedding sources with the respective process. A custom metrics adapter collects the metrics from different spark job process via single channel and store it in a centralized metrics database.

The following features are added in TDH:

- Metrics Collection is a solution for Operational Monitoring the number of events or records processed in each stages of Landing, ODS Process and SDS Process. Also, it captures the number of failure or error data in every stages of the process.
- Metrics dashboard helps to operate and monitor the statistics.

- Spark has a configurable metrics system based library, which allows to report the metrics via variety of sinks.
- GraphiteSink registers the spark process and trigger the metrics details to Graphite node.
- Landing process registers with sink to push the application/entity wise record count through metrics adapter to database and the same is displayed on the dashboard.
- ODS process registers with sink to push the application/entity wise record count through metrics adapter to database and the same is displayed on the dashboard.
- SDS process also leverages the spark metrics sink to push the application/entity wise record count through metrics adapter to database and the same is displayed on the dashboard.

Click here to understand the technical impact of this enhancement for customisation and upgrades.

The topics related to this feature are given below:

Spark Optimisation

Metrics and Operational Monitoring

Transact Data Hub » Extend Livy Integration

This feature allows automatic deployment of Designer and Scheduler jobs using the Livy server. In this feature, the Designer or scheduler spark jobs are submitted using Livy server to Kubernetes cluster and are monitored using batch-ids generated for each job by Livy server.

Users can monitor the job status through Livy UI in the browser and also check the batch logs for the same.

The topic related to this feature is given below:

Integrating Livy in Designer and Scheduler

Transact Data Hub » TDH Packaging

TDH packaging aims to maintain artefacts in one place, which are common across environments, and to deliver the list of applications applicable for Model Bank and Analytics models. As part of package, this feature releases new shell scripts, DDL and DML template scripts for Docker or Non-Docker environments.

Using TDH packaging, users can deploy bulk applications across DES, ODS and SDS seamlessly.

The topic related to this feature is given below:

TDH Packaging

Operational Data Store APIs » Extension of ODS API Landscape

Semantic Query Layer allows you to access the Temenos Transact data from Operational Data Store (ODS) in data lake solution, which will separate the read from the Transact system. As part of this solution, semantic layer will have a framework and exposes the GraphQL end point to accept the GraphQL query as an input and process it to access the data from ODS. This generic framework leverages the GraphQL and SQLBuilder libraries to process the GraphQL query and converts them into SELECT statements to retrieve the data from ODS.

As part of this extension phase, 90 APIs are released from Holding, Party, Reference, Order, and Product domains to access data from ODS in Data Lake.

The topic related to this feature is given below:

Operational Data Store APIs

Banking Framework

Collateral » External Collateral Assets

External assets such as assignments, floating charges, fixed charges and life insurances are not maintained in Temenos Transact core applications. The information related to these external assets may be maintained in bank-specific local tables. These external assets can be provided as collateral for securing a certain facility or liability in Temenos Transact. Temenos Transact has been enhanced to allow users to configure external assets as collaterals, to be processed by the Collateral module.

The topics related to this feature are given below:

External Collateral Assets

Tasks for External Collateral Assets

Collateral » Other Internal Collateral Assets – Immovable Property and Mortgage

Temenos Transact supports assets such as deposit accounts (AD/AR/AC), guarantees (MD), money market deposits (MM) and customer portfolio (SC) as collateral. Users can now configure immovable property and mortgage as internal assets for processing through the Collateral module. Based on the setup defined in the `COLLATERAL.TYPE` application, the system captures all the details with respect to these assets. The `ASSET.REG.PROPERTY` and `CO.REG.MORTGAGE` applications are introduced in Temenos Transact to store the details of the immovable property and mortgage, respectively.

The topics related to this feature are given below:

Other Internal Collateral Assets

Tasks for Other Internal Collateral Assets

Corporate

Misc Deal/Issue Register Guarantee » Capturing Multiple Applicant/Multiple Instructing Party/Beneficiary under the Same Contract

In the `MD.DEAL` application, when registering a guarantee contract, fields related to the applicant and instructing party will now allow user to input more than one customer for all types of contracts other than counter guarantee and/or counter counter-guarantee type of contracts. Such primary applicant and instructing party can be mapped as output in all relevant print advices.

This provision to add or update multiple values for applicant and instructing party in a guarantee contract (for both, issuance as well as on receipt of guarantee), aligns with any type of requirements for guarantee under a Single Trade Agreement involving more than one applicant and/or beneficiaries.

Besides, an applicant can now avail a bank guarantee from the guarantor bank even if there is no line with the bank. The user can issue a guarantee using the credit relationship of another third party customer, i.e. the Instructing Party. This Instructing Party can also be a non-bank entity that can be a parent, subsidiary, associate of anyone of the applicant or the applicant himself.

The topics related to this feature are given below:

Introduction to Issue/Register Guarantee

Working with Issue/Register Guarantee

Misc Deal/Commission and Charges » Calculation and Collection of Commission Values when Commission Period is Set

The following are the new functionalities:

- The *Minimum Commission Rate* field is added to the `MD.TXN.TYPE.CONDITION` table and it represents the percentage of the guarantee amount in `MD.DEAL` for which a minimum commission is arrived. It can be used along with the existing fields for arriving at a minimum commission value to compare with the arrived guarantee commission amount (against a commission rate captured) while entering a guarantee contract in `MD.DEAL`.

  Introduction of minimum commission amount as a percentage of guarantee amount enables the bank to set a higher commission cap for a higher guarantee amount. Also, all aspects of guarantee exposure is covered while setting the minimum commission criteria (that is, calculating based on flat value, currency of guarantee, tenure and guarantee amount).

- When the Commission Period is set, Temenos Transact now supports the following:
  - **Fixed Amount Feature** - Commission can be calculated as a fixed amount per period.
  - **Commission Frequency** - Commission can be collected based on input in CSN frequency. Based upon this frequency, the system generates schedules for collection of commission in periodic instalments.

    Supporting CSN frequency and fixed amount offers the bank different options to collect commission either fully in advance or in periodic intervals, both as fixed amount per period or as a percentage of outstanding guarantee amount.

    Overrides in CSN Frequency allow the user to have better control checks while creating commission schedules

- **Minimum Commission Amount** - Minimum commission amount constraints can be set in the MD.TXN.TYPE.CONDITION table when a commission period is set. Accordingly, the user can set the constraints such that, during the issuance of the contract, the commission amount calculated from MD.DEAL is less than the minimum commission value. Also, the minimum commission value when defaulted from MD.TXN.TYPE.CONDITION, can be collected in periodic intervals based on CSN Frequency or full amount in advance.

  > **NOTE:** Minimum commission calculated herein remains the same for any deals with a given guarantee amount, commission rate and tenure irrespective of whether the CSN Frequency is set or not.

- If both Minimum Commission Amount and Minimum Commission Rate are set, it calculates the greater of the value based on the Minimum Commission Rate (for the guarantee amount entered in MD.DEAL) and the Minimum Commission Amount. The commission is calculated based on the commission rate captured in MD.DEAL for the *Minimum Tenure Value* defined in MD.TXN.TYPE.CONDITION. This is the minimum amount against which the guarantee commission value is compared.

  Enhancing the *Minimum Commission Amount* fields in MD.TXN.TYPE.CONDITION facilitates the banks to collect commissions based on whole periods applicable during the tenure under a guarantee instead of applicable broken periods as practiced by others.

> **The topic related to this feature is given below:**
>
> Working with - Commission and Charges

Private Wealth

Securities/Derivatives » MiFID Transaction Reporting

The 2007 - 08 financial crisis resulted in regulatory authorities worldwide re-considering their approach towards controlling risk within financial markets. Numerous subsequent legislations have been introduced to achieve a more efficient and transparent financial system. Markets in Financial Instruments Directive (MiFID) II is the most substantial post-crisis regulatory change within the European Union. MiFID II has strict requirements in terms of transaction reporting as well as pre and post-trade transparency. Investments firms which execute transactions for customers are obliged to report the transactions to the competent authorities not later than the end of the next working day that is, T+1 day. The investment firms use the intermediary Approved Reporting Mechanism (ARM) to submit the reports to the competent authorities.

Temenos Transact is enabled to report all the transactions to oblige the MiFID II - Markets in Financial Instruments Regulation (MiFIR) regulation. The benefits of this feature are that:

- The transactions input in the system can be reported along with all the required details as part of MIFID transaction reporting.
- The reports can be submitted in any format such as csv, text, and so on based on the setup.

The topics related to this feature are given below:

Securities – MiFID Transaction Reporting

Derivatives – MiFID Transaction Reporting

Regional Banking Solutions

Argentina Model Bank

Accounts » Embargo Process

The Embargo Process functionality allows the processing of different trades from COELSA, which manages all the trades received from AFIP (the tax revenue agency of Argentina) on the defaulted customer. The two main steps in the seizure process through COELSA is an online and batch process.

The Balance Enquiry API call is deprecated and the Lift of Seizure API is included.

The *Release Date* and *Release Amount* fields are added to the `EMBARGO.DETAILS` application.

> **The topic related to this feature is given below:**
>
> Accounts

Accounts » Pre-closure Fees During Cooling Period

This functionality allows banks to set up the cooling period during which the pre-closure fees won't be applied, to the customer.

The cooling period is defined in the closure product condition according to the Argentinian regulation as 10 working days. The pre-closure fees won't be applied if the payoff is done on or before the cooling date set in the *Cooling Date* field from the `AA.ACCOUNT.DETAILS` application.

> **The topic related to this feature is given below:**
>
> Accounts

Taxes » Perception VAT

This functionality allows banks to apply two different default base tax rates for Perception VAT (national tax, calculated over the added value in purchased operations in products or services) for taxpayers flagged in the AFIP Contributor file as responsables inscriptos, depending on the tax event.

For customers flagged as “responsables inscriptos” in AFIP Contributor file (where the *Imp Iva* field is equal to AC), Temenos Transact will post a second VAT transaction on the customer account.

This second transaction will have a separate base tax rate depending on the tax events:

1. The base default rate for Perception VAT calculated over the debit interests of a loan, will be 1,5%.
2. The base default rate for Perception VAT calculated over other tax events, like fees on saving accounts, will be 3,00%.

To determine the final rate, the RG17 Padron file will be checked and, if there is any reduction, it will be applied over the base tax rates specified above.

MONOTRIBUTO specifications for VAT will only have an impact on the Perception VAT, not on the generic VAT.

> **The topic related to this feature is given below:**
>
> Taxes

Australia Model Bank

Australia - Base » Alternate Account Number Generation and User Transaction Restriction

The Alternate Account Number Generation functionality allows banks to automatically generate an alternate account number for each account as per the bank configuration.

The User Transaction Restriction functionality prohibits bank users from crediting or debiting their accounts.

The `CMBASE.ALTERNATE.ID.PARAM` parameter application is used to configure the way alternate account numbers are generated during arrangement creation.

The *User Customer Number* field is added to the `USER` application and it is used to manually capture the *Customer Number* of the bank user.

The system will check whether the customer of the account (arrangement) is the same as the *User's Customer Number* stored in the `USER` application. If it is the same, then the system will not allow the user to commit credit or debit activity.

> **The topic related to this feature is given below:**
>
> Australia - Base

Withholding Tax and Trust Income Distributions » Accrued Tax Based on Accrued Interest

This functionality allows banks to use in their reporting the amount of withholding tax due for a customer on the accrued interest (not yet capitalized) on a given day.

The XAUADDITIONAL.INFO external property class is introduced to hold the

Accrued Tax amount.

> **The topic related to this feature is given below:**
>
> Withholding Tax and Trust Income Distributions

Withholding Tax and Trust Income Distributions » Resident and Non-Resident Tax Calculations

As per the Australian Taxation Office (ATO) guidelines, in an event, the customer earns income in the form of interest. The bank must report the earnings to ATO. If the customer fails to furnish certain details like Australian Business Number (ABN), Tax File Number (TFN) mandated by ATO, the Withholding Tax (WHT) will be applied to the interest earned by the customer. The application of WHT rates depends on certain criteria like the age and residence of the customer.

This functionality allows banks to calculate the tax on the total interest when there are resident and non-resident joint-owners on any account that earns interest.

> **The topic related to this feature is given below:**
>
> Withholding Tax and Trust Income Distributions

Finland Model Bank

Collateral » Automatically Delink the Collateral when Liabilities End

This functionality allows banks to automatically delink the collateral once the loan is closed and to update the status of the collateral related document collected from the customer.

The following items are introduced as part of this functionality:

- The `COLLATERAL,DOC.STATUS` version is used to record the details of collateral submitted by a customer, its value(s), to update the status of the collateral related document collected from the customer and, optionally, to link it directly to a customer's supporting deal elsewhere in the system.
- The COLLAT.EXPIRY.DOC.STATUS enquiry allows users to view all the expired collateral records and to change the status of the collateral document for the expired collateral records.
- The *Document Status* field is added to the `COLLATERAL.TYPE` application to configure the default document status value for the collateral type.

> **The topic related to this feature is given below:**
>
> Collateral

Global Model Bank

ATM Framework » Fast Funds Requests - Debit Cards

This functionality enables banks to handle fast funds authorization requests received from merchants using the Visa Fast Funds service, and the transactions for debit cards.

The following items are introduced as part of this functionality:

- The *Ff Amount Limit* and *Ff Suspense Account* fields are added to the `ATM.PARAMETER` application to define the fast funds amount limit and the suspense account to be debited.
- New configuration records are released to map the authorization request and response and to define the processing code needed for fast funds transactions.

> **The topic related to this feature is given below:**
>
> ATM Framework

Hungary Model Bank

Warrants » Locking of Accounts

This functionality allows banks to create manually locks in Temenos Transact.

The following items are introduced as part of this functionality:

- The *Lock Priority* and *Lock Type* fields are added to the `HUWRNT.QUEUE.PARAMETER` application to define the lock types and their priority.
- The `HUWNRT.MANUAL.LOCKS` live application is updated only when the lock created is for priority 3 so that the locks can be used by the Queuing solution.
- The `AC.LOCKED.EVENTS,HUWRNT.OTHER.LOCKS.INPUT` version is used to create locks manually.

> **The topic related to this feature is given below:**
>
> Warrants

India Model Bank

Lending Compliance » Structured Financial Messaging System (SFMS)

This functionality allows the user to capture the Indian Financial System Code (IFSC) codes of the banks and generate INFINET (Indian Financial Network) Format Number (IFN) messages whenever a transaction happens in the LC (Letter of Credit), BG (Bank Guarantee) or Bills application, that warrants the exchange of messages between two banks or branches within India.

The bank or branch within India for whom the IFSC code is to be captured is created as a customer in Temenos Transact following an entry in the `DE.ADDRESS` application and also in the `RD.CENTRAL.BANK.DIR` application.

The `DE.MAPPING` application channels the SFMS messages based on the IFSC bank code.

The `CUSTOMER,INLEND.BANKBRANCH.INPUT` version is introduced to hold the basic minimum details that are required to create the customer record.

The topic related to this feature is given below:

Lending Compliance

Lebanon Model Bank

Salary Blocking

This functionality allows banks to safeguard loan payments by locking the instalment amount once the salary is credited to the settlement account.

The following items are introduced as part of this functionality:

- The SALBLK.SAL.BLOCK.PARAM application is used to define which customer, product, category etc. can be exempted from salary blocking and allows users to define which property has to be configured for calculating the salary block amount.
- The SALBLK.SAL.BLOCK.DIFF.CURR enquiry is used to list all the locked amounts when the currency of the salary is different from the currency of the loan.
- The SALBLK.SALARY.LOCK application is used to store the lock amount details.
- The SALBLK.SALARY.LOCK.HIS application is used to store the lock amount history details.

> **The topic related to this feature is given below:**
>
> Salary Blocking

New Zealand Model Bank

Customer and Account Infrastructure » Account Number Validation

All New Zealand bank accounts have to conform to the Payments New Zealand (PNZ) bank account numbering convention or standard which is subject to two modulus check – module 11a and module 11c.

This functionality allows banks to validate the pay-in or pay-out account number during the arrangement creation based on the PNZ (Payments New Zealand) format.

The `NZBASE.PARAMETER` parameter application and the `NZBASE.PARAMETER,INP` related version are introduced as part of this functionality. The pay-out number will be validated against the parameters defined in this application.

The topic related to this feature is given below:

Customer and Account Infrastructure

Norway Model Bank

Lending » Adjust the Interest Rate When a Loan is Changed before Maturity

This functionality allows banks to offer a dynamic Fixed Rate Agreement which can be either 3 years, 5 years, 7 years, or 10 years. The bank can extend the Fixed Rate Agreement period as desired. Banks can also calculate the premium or the discount when the fixed-rate agreement is breached.

The Norway Consumer Lending product is added to the Norway Lending product group to allow banks to create loans, perform the required changes, achieve the levying of the premium or the discount for the customer during loan modification like changing the customer details, changing the interest rate, loan prepayment and loan pre-closure (payoff).

The following applications are introduced as part of this functionality:

- The `NOLEND.PARAMETER` application allows the bank to configure the agreement change period, the interest property for the NOLEND product group.
- The `NOLEND.CASHFLOW.DETAILS` application allows the bank to store the charge details of the loan arrangement created as a Fixed Rate Agreement.

> **The topic related to this feature is given below:**
>
> Lending

Qatar Model Bank

Account Infrastructure » Customer Info Capture and Regulatory Codes

This functionality allows banks to capture customer data and information relating to corporates (and other business entities) as required by regulatory authorities.

New configuration records are released as part of this functionality to capture the customer data and the information relating to the corporates (and other business entities) as required by the regulatory authorities.

> **The topic related to this feature is given below:**
>
> Account Infrastructure

Account Infrastructure » Dormant Accounts

This functionality allows banks to set up the dormancy statuses and to extract reports containing the accounts with various dormancy statuses and details of the transactions that happened in an internal account when the account is closed due to unclaimed status.

This functionality allows banks to apply posting restrictions on accounts when the accounts are in dormancy status.

The following enquiries and versions are introduced as a part of this functionality:

- The QAACIN.DORMANCY.REPORT enquiry lists the accounts that are going to become dormant in one month, by giving the status as defined by the bank. The bank is allowed to use this enquiry, to fetch the details of the account based on the dormancy status.
- The QAACIN.MAINTENANCE.REPORT enquiry displays the customer's details for the accounts with the status manually changed to Active.

- The QAACIN.UNCLAIMED.GL enquiry lists the accounts that are dormant and whose unclaimed balances have been moved to unclaimed balances GL.
- The `QAACIN.DORMANT.PARAM,INPUT` version allows users to define the posting restriction for a particular dormant status.

> **The topic related to this feature is given below:**
>
> Account Infrastructure

Saudi Model Bank

Early Closure Charges » Customer Deposits

This functionality explains the solution related to how the rates used for the staff deposit and collateral are changed when the deposits are not for staff and the collateral is no longer attached to the deposits.

Banks can open deposit accounts for customers and also for its staff. There are various types of deposits, the deposits may vary based on the tenor, interest rate, amount, etc. Based on the customer, amount, tenor etc. the interest rate can be different. The bank will also provide facilities like partial withdrawal or early withdrawal or pre-closure i.e. when the whole amount or the partial amount are withdrawn before the maturity date. The deposit amount can be renewed when the maturity date is reached.

The details of the treasury reference and dealer code for special rate can be captured in the *Treasury Ref No* and *Dealer Code* fields.

The various types of products can be configured in Arrangement Architecture based on the products that the bank offer to its customers and staff.

The topic related to this feature is given below:

Early Closure Charges

Hijri Date Conversion » Standing Orders

Banks in Saudi follow the Hijri calendar and not the Gregorian one. This functionality allows banks to create a standing order using the Hijri date. They can also stop or disable a standing order (STO) for the customers whose legal IDs have expired.

The *Hijri Date* field is added to the `STANDING.ORDER, HIJIRI.INPUT` version to allow users to create a standing order using the Hijri date.

The *Hijri Start Date* field is added to the STO application to display the *Start Date* using the Hijri calendar.

The *Spl Gr1 Days Aft Exp* field in the SABASE.CUSTOMER.PARAM application is used for parameterising the grace period for standing orders after the legal id has expired (in days).

The SAHJRI.HIJRA.TO.GREG service is used to disable the standing orders for which the initiating customer's legal ID is getting expired in the current date.

The *No. Of Months for Expiry* field was added to the SAREGS.PARAMETER application to store the maximum number of months an expat can hold a standing order, for example 6M, 12M.

> **The topic related to this feature is given below:**
>
> Hijri Date Conversion

SIMAH Credit Bureau Interface » Handling Loan Reversal

This functionality allows banks to report the files already reported and reversed to Saudi Credit Bureau (SIMAH) with the status set as cancelled.

The *Reversal Record* field is added to the SASIMA.SELECT.CONCAT application to store the XML file generated when a new loan is created.

> **The topic related to this feature is given below:**
>
> SIMAH Credit Bureau Interface

Spain Model Bank

Cheques and SNCE Clearing » SNCE08 Improvements

This functionality allows banks to process the SNCE08 message for SEPA (Single Euro Payments Area) Credit Transfer (SCT) outward and inward requests, and for SEPA Direct Debit (SDD) outward and inward requests for the transaction types 10 and 14.

The following items are introduced as part of this functionality:

- The `ESCLNG.SNCE.DATA.MODEL,DONE` version is used to change the status as Done for the record types 70 and 30.
- The ESCLNG.SNCE08.BO.ACTION enquiry is used to change the status of the requests for the record types 30 and 70.
- The ESCLNG.SNCE08.CT.RECALL enquiry is used to initiate the cancellation requests.
- The ESCLNG.SNCE08.DD.RECALL enquiry displays only those DD's that are out of the SEPA regulated period (domestic).
- The ESCLNG.SNCE08.INCOMING.ACTION enquiry is used to accept, reject, or return the inward requests.
- The ESCLNG.SNCE08.INCOMING.CT.DD enquiry is used to view all the inward requests.

> **The topic related to this feature is given below:**
>
> Cheques and SNCE Clearing

Tunisia Model Bank

Foreign Currency Operations » F1 or F2 Authorization Request

This functionality allows banks to store the details of the F1 or F2 sheet, which has to be sent to the Central Bank for authorisation to proceed with the prohibited transactions.

The following applications, versions, and enquiries are introduced as part of this functionality:

- The `TNFCOP.FORM.APPLICATION` application is introduced to store the information from the customer and update it in Temenos Transact for the F1 or F2 applications. If an amendment is required, it needs to be done by the branch and the regulatory control and they need to update the additional information requested by CBT or regulatory control.
- The `TNFCOP.FORM.APPLICATION,DOMICILIATION` version is used to create F1 or F2 sheets.
- The `TNFCOP.FORM.APPLICATION,AMEND` version is used to amend the details of the F1 or F2 sheet by adding additional information.
- The `TNFCOP.FORM.APPLICATION,RESUB` application is used to resubmit the F1 or F2 applications in case of any changes required.
- The `TNFCOP.FORM.APPLICATION,DECISION.UPDATE` application is used by the Regulatory Control Team to update the status of the application.
- The `TNFCOP.FORM.APPLICATION,RES.F1F2` version is used to manually update the reservation of the forms.
- The `TNFCOP.FORM.APPLICATION, SETT.F1F2` version is used to manually update the settlement of the forms.
- The `TNFCOP.FORM.APPLICATION,CANCEL` version is used to cancel the F1 or F2 sheet created and submitted to the Central Bank.
- The TNFCOP.REPORT.STATUS enquiry is used to fetch the status of the form and prepare an excel file or PDF file which has to be shared with Central Bank.

The topic related to this feature is given below:

Foreign Currency Operations

Foreign Currency Operations » Information Sheet

This functionality enables banks to create the information sheets for making any payments in foreign currency and report to the Central Bank of Tunisia for the services opted by the customers.

The following applications, versions, and enquiries are introduced as part of this functionality:

- The `TNFCOP.INFO.SHEET` application allows banks to store the application details of the information sheet, which has to be approved by the bank and to be reported to the Central Bank of Tunisia.
- The `TNBASE.FINANCIAL.DETAILS` application is introduced to store the details of the total utilised amount and reserved amount for the information sheet.
- The `TNFCOP.INFO.REPORT` application allows banks to store the period for which the information sheet needs to be generated.
- The `TNFCOP.INFO.REPORT,GENERATE` (Generate New Report) version allows banks to define the range date for which the report will be extracted.
- The `TNFCOP.INFO.REPORT,REGENERATE` (Regenerate Report) version allows banks to define the range date for which the report will be regenerated.
- The TNFCOP.SERVICE.REQUEST enquiry allows the branch user to access the `TNFCOP.INFO.REPORT,GENERATE` (Generate New Report) and `TNFCOP.INFO.REPORT,REGENERATE` (Regenerate Report) versions.

The topic related to this feature is given below:

Foreign Currency Operations

Retail

Islamic Banking » Advanced Settlement of Bills – Upfront Profit Arrangements

In upfront profit contracts with annuity type of payment schedule, the customer can opt to pay a few installments in advance and skip as many repayments that are scheduled. When the customer makes an advanced repayment for one or more upcoming bills, based on the configuration, the income is either booked to the P/L immediately or it is amortised over the advance repayment period (respite period).

When the advance repayment is amortised over the respite period, the regular accrual for the respite period is stopped. The system recalculates the profit rate for the respite period and accrues the income to the P/L on a daily basis until the end of the respite period. After the respite period, the accrual continues on the outstanding principal after the respite.

When the income is booked to the P/L immediately on the respite date, there will be no bills made due for those upcoming number of installments for which the advance repayment was done as they are settled. The settled principal amount is reduced from the outstanding principal and the accrual continues from the next day on the outstanding principal. Similarly, the income received is received from the receivable income for the upfront contract. The profit rate is recalculated for the profit pending to be received from the customer.

The configuration for the above scenario is done in the *Advance Profit Accounting* field in Interest product condition. The values that can be set are:

- **Amort** – The income received gets booked to the PL over the respite period.
- **Cash Basis** - The income is credited to the PL on the same day of receipt of the income (on the respite date) and the accrual continues on the outstanding principal.
- If no option is selected, the field value is considered as Null and the system functions as per the functionality applicable for Cash Basis.

The topics related to this feature are given below:

Accounting during Respite period

Configuring Finance – Profit Upfront Sale Product Group

Transaction Recycler » Defining Recycler Priority for Bill-Type wise Settlement

The Transaction Recycler (RC) module captures transactions which cannot and should not be completed and retry them until they are completed or abandoned.

When there is more than one bill pending for settlement against a settlement account, it is possible to sort them based on the aging status, bill date or the bill amount. It is now possible to prioritize bills pending for settlement based on the bill type as well.

The *Rank Info* field in `RC.DETAIL` is updated based on the sorting priority defined against Priority Rank Type in `RC.CONTRACT.PRIORITY`. When the *Priority Rank Type* is set as:

- Blank - The *Rank Info* is updated with aging status of the bill.
- Aging Status - The *Rank Info* is updated with aging status of the bill.
- Bill Type - The *Rank Info* is updated with the bill type.

The topics related to this feature are given below:

Transaction Recycler

Configuring Transaction Recycler Priority

Contract Priority to Prioritise Bills

Technology

Integration Framework

Integration Framework Designer » Conditional Event Generation for Banking Framework Events

The Conditional Event Generation (CEG) feature has been added to Banking Framework events. This means in event designer it is now possible to define filters for the events to be generated for Banking Framework events.

> **The topic related to this feature is given below:**
>
> Using Conditional Event Generation

Integration Framework Designer » Enabling or Disabling Flows

Event Framework generates business events for all the flows defined for an exit point. The only way to stop event generation is to delete the flow and update the relevant exit point from Transact. To avoid deleting the flow, a new feature has been introduced, which helps to disable the flow and thus prevents unnecessary events being generated in Event Framework. This feature is a configurable option in the `IF.FLOW.OVERRIDE` table.

Using this feature, a user can:

- Stop triggering events for a flow instead of deleting the flow.
- Enable and disable flows.

The topic related to this feature is given below:

Editing a Flow by using a Temenos Transact Application

Interaction Framework

IRIS R18 » Automatic Generation of Publisher APIs in IRIS

The new feature in IRIS automates the generation of Publisher APIs. It also provides the solution to generate the Swagger and `service.xml` during design time by uploading an existing Swagger.

This new feature directly maps the Provider URL that is obtained by the user in the `service.xml` for each endpoint along with the selection mapping. Users can download both Swagger and `service.xml` as a zip file from Workbench.

The topic related to this feature is given below:

Automatic Generation of Publisher API

Platform Framework

Adapter Microservice

Temenos Microservices architecture allows transact and infinity modules to develop independently scalable Microservices. To achieve independent scaling, it is recommended to use the events-based approach instead of typical synchronous or file-based integration.

To accomplish any business functionality, Temenos business microservices may have a dependency on other applications in the following scenarios:

- The third-party system provides an API based interface for executing a business action or service.
- The third-party system provides only a file based interface for sharing data.

To keep business microservice independently scalable, Adapter Microservice has been introduced. It provides a scalable interface to meet the above integration-related requirements. It also provides the flexibility to deploy API specific transformation and routing configurations for interface.

The topic related to this feature is given below:

Adapter Microservice

Treasury

Forex » Utilisation of FX Contracts

Corporates hedge payables and receivables of their foreign currency and conversion to the local currency by entering into Foreign Exchange (FX) contract with the bank's treasury. The FX module is enhanced to:

- Capture FX deals without settlement instructions for Transaction Processing Application (TPA) to utilise these FX contracts. The settlement happens in TPA, which enables to generate accounting entries.
- Support automatic utilisation of FX contracts by other TPAs (such as Loans, Payments, and Trade Finance) in Temenos Transact.
- Update FX contract balance on real-time basis

This benefits the user to utilise the FX contract from other departments without any manual intervention.

Click here to understand the technical impact of this enhancement for customisation and upgrades.

The topics related to this feature are given below:

Introduction to Facility to Utilise FX Contracts

Forex Reports and Enquiries

Tasks for Facility to Utilise FX Contracts

Enquiries and Reports

I Technical Notes

Analytics

Transact Data Hub » Spark Optimisation, Metrics Collection and Operational Monitoring

Ensure you import designed JSON files in Grafana to integrate and display the metrics information in dashboard.

Banking Framework

New Module – Bulk Payment Initiation

A new module, Bulk Payment Initiation (BU), has been split from the Funds Transfer (FT) module.

`FT.BULK.MASTER` and `FT.BULK.UPDATE.TYPE` applications have been moved from FT to BU module.

- This module will be automatically installed during the upgrade process and requires no manual intervention.
- Any local developments referring to the `FT.BULK.MASTER` and `FT.BULK.UPDATE.TYPE` tables in componentised format should be modified to refer BU component.

Treasury

Forex » Utilisation of FX Contracts

A new Close of Business (COB) service (FX.EXT.SETTLE.REPORT) is available to generate a report to list the FX transactions, which have a residual balance on the value date. Hence, it can be settled in the suspense account defined in the *Residual Amt Settl* field of `FX.PARAMETERS`.

Extensibility APIs

Technology

Java Extensibility

The extensibility APIs for Java are:

#	Package	Class	Method name	Exten-ded As	Extens-ibility Category	Description
1	payments	`PaymentLifecycle`	`postUpdateRequest`	Hook	New	Defines a request to update core records to be posted asynchronously.
2	payments	`PaymentLifecycle`	`postRequestToExternalSystem`	Hook	New	Decides whether to post a request to an external system using the returned boolean value.
3	payments	`PaymentLifecycle`	`updateRequestToExternalCoreSystem`	Hook	Deprecated	Deprecated since most of the parameters are made as INOUT from IN and codes are not as per naming convention mentioned in the L3 governance guidelines. Use `PaymentLifecycle.postRequestToExternalSystem` instead.
4	system	`Session`	`publishMessage`	API	New	Transforms the record, publishes it as per the configuration and returns the published message.
5	system	`Session`	`printLine`	API	New	Prints the message to the Temenos Transact standard

#	Package	Class	Method name	Extended As	Extensibility Category	Description
						output, prepending the process name, job name, session, date and time details.
6	system	DataAccess	getFieldValue	API	New	Returns the entire value of a Temenos Transact field given the application name, field name and record.
7	arrangement.accounting	Contract	getNextPayment	API	Existing	Fetches the next payment date and next payment amount of an arrangement.
8	system	Session	setNextVersion	API	New	Enables the developer to define a VERSION that will be launched on successful commit of the current transaction. (Enables EB.SET.NEXT.TASK for VERSION)
9	system	RecordLifecycle	enableAutomaticAuthorisation	Hook	New	Enables the developer to write code to identify whether transactions can be authorised automatically. (Enables the EB.STP.CONDITION functionality)
10	payments	PaymentLifecycle	updateProduct	Hook	Existing	Helptext change to accommodate heavy weight product updation as well.