What's New in

Temenos Transact

February 2021

Table of Contents

Release Highlights

Analytics

Semantic Query Layer » Data Enrichment in Semantic Query Layer

The following features are updated in the Semantic Query layer (SQL) module:

- SQL supports enrichment rules that enriches the resultant data by applying the rules defined for APIs accessed at runtime. The SQL framework enables the below enrichment rule features to design and evaluate the rules for APIs:
  - Replace
  - Alias
  - Concatenation
  - Calculation
  - Condition
  - Substring
  - Split
- SQL uses a new caching mechanism EhCache instead of Ignite, which eliminates the external server dependency retaining the existing caching functionalities (API Metadata, GraphQL Schema, DML and API Metrics caching).
- SQL uses Local API Metadata for accessing an API at runtime, thereby eliminating dependency on Atlas for pre-defined APIs.

  NOTE: Atlas is still required for designing the APIs.

Click here to understand the installation and configuration updates for this enhancement.

The topic related to this feature is given below:

Semantic Query Layer

Application Framework

Security Management System » Creating Policy with Enhanced Obligations

Enhanced obligations provide a greater flexibility in designing the obligations. With this feature, the policy editor can:

- Create policies with multiple filter conditions inside a single obligation group.
- Define the grouping operator between the filter conditions.
- Define the grouping operator between the obligation groups.

You can create a policy with enhanced single or multiple obligation groups.

The topics related to this feature are given below:

Design-time

Creating Policy

Banking Framework

Collateral » Collateral Allocation based on Percentage Cap

Temenos Transact is enhanced to support collateral allocation based on the percentage capping of outstanding amount of the Limit. Banks can indicate the percentage cap on the outstanding limit, which is covered by the attached collateral, and allocate the collateral amount (execution value) considering the percentage cap. The *Limit Cap Perc* field is added in the `COLLATERAL.RIGHT` application to capture the percentage cap to be applied on the outstanding amount of the Limit. If the user does not define any percentage cap for a given outstanding Limit, then 100% of that outstanding Limit is considered for collateral allocation.

This functionality provides banks the ability to capture the percentage contribution of the individual collateral provider to the associated liability.

> **The topic related to this feature is given below:**
>
> Collateral Allocation based on Percentage Cap

Cheque and Draft Issue Management » Tax on Cheque Issue Charges

Some banks collect value-added tax (VAT) on all charges and interest from the customer at a rate defined by the local government. Cheque issue charges are one of the charges collected by the bank. This is a variable charge based on the number of cheque leaves issued by the bank to its customers. This cheque issue charge is subjected to additional tax amount at a specific percentage of the charged amount and is collected from the customers.

The *Tax Code* field has been added in the `CHEQUE.CHARGE` application to

apply a tax (such as VAT) to the charges calculated for the number of cheques issued to a customer. The *Tax Code* field accepts a valid TAX or TAX.TYPE.CONDITION record:

- When a TAX record is attached, the system calculates the tax amount as per the TAX record and posts accounting entries when the cheque is issued.
- When a TAX.TYPE.CONDITION record is attached, the system determines (based on the customer grouping conditions from the TAX.GEN.CONDITION record) whether the customer falls under a group and calculates the tax amount as per the TAX record linked in the TAX.TYPE.CONDITION record for that group, then posts accounting entries when the cheque is issued.

The *Tax ID* and *Tax Amount* fields are added in the CHEQUE.ISSUE application to display the tax code and calculated tax, respectively.

> **The topic related to this feature is given below:**
>
> Tax on Cheque Issue Charges

Direct Debit » Processing of Creditor Driven Direct Debit E-mandates

The Direct Debit (DD) module in Temenos Transact provides banks the feature to register and maintain DD mandates manually, through APIs or as an auto-registration process as part of DD collections processed through Temenos Payments Hub (TPH).

A Centralised Mandate Service is a dedicated service offered by the national or regional clearing system. The banks collect new mandate, mandate amendment and mandate cancellation requests from their creditors or debtors and, under this service, send these requests to the Centralised Mandate System directly or indirectly, through a direct participant of the service. The Centralised Mandate

System performs the validations and forwards the valid requests to the instructed agents. The exchange of the messages between a direct participant and its indirect participants is done outside of this service.

Under the licensable feature DD E-mandate (DDEMAN), the DD module provides a framework to a debtor bank to exchange ISO20022 mandate management messages with a Centralised Mandate System for creating or maintaining Direct Debit Mandates. SEDA is a SEPA-compliant Electronic Database Alignment offered to banks in Italy by the STEP2 clearing. The SEDA-specific setup is also released as part of the product.

This functionality allows banks to:

- Receive files containing ISO20022 pain.009, pain.010 and pain.011 messages from the Centralised Mandate System based on which the system will register, amend or cancel DD mandates.
- Send pain.012 acceptance reports back to the Centralised Mandate System confirming the acceptance of the DD mandates registration, amendment and cancellation requests.
- Receive file containing ISO20022 pain.012S2 message for registering the response from Centralised Mandate System to the requests' processing.

> **The topics related to this feature are given below:**
>
> Processing of Creditor Driven Direct Debit E-mandates
>
> Support for E-Mandates

Arrangements Architecture » Defining Mandate Requirements Centrally

In Temenos Transact, the user can set mandates for customers or accounts by directly specifying the mandate conditions applicable to various transactions in the `CUSTOMER` or `ACCOUNT` application. The mandate conditions define the minimum number of signatories that are required to authorise a transaction of a customer or account for various amount ranges.

Temenos Transact now allows the banks to define logical grouping of financial or non-financial applications that perform transactions for which mandate processing is applied. The Mandate Requirement application allows the user to define the mandate conditions applicable to the specified mandate application groups and to indicate if these apply to the portfolio, account or customer.

Temenos Transact provides customers the flexibility to use the new centralised way of defining mandate requirements or continue with the existing method of defining mandate requirements. The option to define mandate requirements at the portfolio level is only supported by the centralised Mandate Requirement application. This functionality provides banks the ability to create and maintain mandate requirement centrally without implying changes for each customer or account.

The topics related to this feature are given below:

Defining Mandate Requirements Centrally

Configuring `EB.MANDATE.PARAMETER`

Configuring EB.MANDATE.APPLICATION.GROUP

Mandate Requirement

Mandate Processing

Delivery » Emitting Delivery Messages to Event Store

Infinity products can manage and process alerts for Infinity users by consuming events from an external system. The Delivery (DE) module in Temenos Transact emits delivery events details only to Temenos Transact customers.

The Delivery module has been enhanced to provide the ability to emit delivery messages or events to the Event Store microservice, which can be consumed by Infinity products or other external systems.

The topic related to this feature is given below:

Emitting Delivery Messages to Event Store

Account Reporting Events and Services » Entry Event for Funds Transfer

The Account Reporting Events and Services (IZ) module in Temenos Transact has been enhanced with an additional Entry Event for streaming the accounting entries specific to the `FUNDS.TRANSFER` application.

This functionality enables Preconfigured Entry Event specific to the `FUNDS.TRANSFER` application, which provides more transaction-specific enrichment to the Entry and Transaction Detail tags for Cash Management (CAMT) messages.

The topics related to this feature are given below:

Out-of-the-box Event Data Flows

System Tables » Generating Accrual and Amortisation Entries through Soft Accounting

The `EB.ACCRUAL` application is enhanced to use the Temenos Transact Soft Accounting functionality to generate the daily accounting entries for all AA Arrangement contracts. This allows banks to use rule-based accounting to generate the daily accrual or amortisation entries for AA Arrangement contracts.

Click here to understand the installation and configuration updates for this

enhancement.

The topic related to this feature is given below:

Configuring Soft Accounting

Accounts » Configuring Incoming MT920

Account Statement functionality supports Outgoing and Incoming (MT920) Statement request in Transact through `DE.STATEMENT.REQUEST` application. Even though it supports both the Outgoing and Incoming (MT920) Statement requests, the configuration was available only for the generation of Outward statement request.

To overcome the above, the Account Statement functionality is enhanced to configure Incoming MT920. It provides a single option to manage the Incoming statement request to view, edit and to action further.

It enables you to view the list of authorised and unauthorised incoming statement request records from `DE.STATEMENT.REQEUST` application and perform the following processes:

- View and amend the Account setup using `ACCOUNT.STATEMENT` application for which the MT920-statement request is received.
- View, amend and authorise the incoming statement request through `DE.STATMENT.REQUEST` application
- View the generated Outward Swift Message (Statement Response) for the corresponding incoming statement request
- View the incoming swift message.

The topics related to this feature are given below:

Tasks for Account Statements

Enquiries and Reports

Delivery » Delivery as Microservice Events

Delivery (DE) module in Transact is enhanced with the ability to emit delivery messages or events to the Event Store microservice, which then can be consumed by Infinity or other external systems by configuring a dedicated delivery interface to the delivery carrier. This feature enables the bank administrator who may or may not have `CUSTOMER` record in Transact, to raise an alert request the account/arrangement to which he is not associated at the Transact level. Transact provides a placeholder (*Ext User Id*) in `EB.ALERT.REQUEST` for the bank to specify the user id in the alert request. This is a free text field and the validations are done by the Infinity system when a subscription request is raised to Transact.

The topic related to this feature is given below:

Configuring Soft Accounting

Private Wealth

Securities » Valuation and Average Price Calculation in FIFO Method

Regulations in some countries require that certain own book and customer portfolios has to be maintained on a First in First Out (FIFO) valuation basis. Temenos Transact supported only the weighted average cost valuation and average price calculation for all portfolios.

In order to support regulatory requirements, Temenos Transact now performs the valuation and average price calculation based on FIFO method for the customer and own book portfolio.

The topics related to this feature are given below:

FIFO based Valuation and Average Price Calculation for Customer and Own Book Portfolio

Own Book Portfolio with FIFO Valuation

Security Position Details of Portfolios with FIFO Valuation

Securities » Determining the Place of Settlement and Defaulting in the Transaction

The Place of Settlement (PSET) is the final depository where the transaction settles. PSET is one of the key to determine the counterparty settlement instructions (SSI) and is mapped in the settlement SWIFT messages that is sent to the custodian (MT 540-543).

PSET is now determined using a combination of several keys namely, clearing code, country code, product type, product sub-type, stock exchange and defaulted in transactions. This functionality helps the banks to determine the

correct PSET based on several keys and default it in the transaction.

The topic related to this feature is given below:

Defaulting the Place of Settlement in the Transaction

Securities » Determining NCI for MiFID II

The European council has provided detailed regulations to identify natural personal based on their National Client Identifier (NCI). The NCI of all individuals involved in a transaction (that is, when an investment is bought or sold) must be checked and reported to the regulators by the bank as part of transaction monitoring and reporting. The nationality of the customer is used to determine the NCI applicable for the country. The bank has to identify the appropriate NCI for the customer based on the priority applicable for the country. Different countries use different identifiers as NCI and there could be multiple NCIs and the most appropriate NCI has to be reported based on their priority.

The NCI parameter is introduced in Securities module to enable NCI determination.

- The `SC.NCI.PRIORITY` application holds the applicable NCIs per country in order of priority.
- If an individual has not provided an NCI or in some countries where they do not want to rely on an NCI document, a concatenated value is reported. This concat code is built by combining the nationality, date of birth, first name and surname of the individual. The system builds this concat code as required.
- Additional fields are provided in `SEC.TRADE`, `SECURITY.TRANSFER`, `SEC.OPEN.ORDER`, `SC.BULK.TRANSFER`, `POSITION.TRANSFER`, `SY.DCI`, `FOREX`, `ENTITLEMENT`, `DX.ORDER` and `DX.TRADE` applications to hold the NCI code of all the stakeholders involved in the transaction such as the ordering customer, the decision maker, instruction maker, trader and manager. The LEI and NCI codes are updated from the Wealth

Suite Front Office (WSFO) system (TAP) or can be interfaced from any front-end system. In case of transactions directly input in Temenos Transact, then Temenos Transact updates these values.

NCI reporting is a key regulatory requirement of MiFID II and this functionality helps in complying with the same.

The topics related to this feature are given below:

Determining NCI for MiFID II

Stock Exchange

Setting up Customer Account

Calculation Formula and Frequency Basis

Setting up Portfolio for Reimbursement

Tasks for NCI Determination for MiFID II

Enquiries and Reports

Defining SC National Client Identifier (NCI) Parameter

Regional Banking Solutions

Argentina Model Bank

Accounts » UVA Deposits

There is a new type of regulatory deposit in Argentina, an early redemption deposit with Purchasing Value Unit (UVA) or Housing Unit (UVI) values. In Argentina, two indices are used for loans and deposits. Since its creation at the beginning of 2016, the financial system delivered mortgage loans or deposits in UVA or UVI. The UVA value is updated daily by the Reference Stabilisation Coefficient (CER). It is published by the Central Bank of Argentina (BCRA) on its website.

This new regulatory follows the new macroeconomic context of Argentina of low taxes and it tries to stimulate the investments in local currency offering an option that adjusts to the evolution of the prices.

The purpose of this functionality is to give a new fixed term deposit with early redemption option following the mandatory application of Financial Entities that offer any other type of fixed term deposit. This new pre-cancellable type of deposit will have 2 terms, 90 and 180 days.

The topic related to this feature is given below:

Accounts

Taxes » Exemption Framework

According to Argentinian regulation there are several exemptions and special conditions related to the calculation of different taxes. They can depend on the jurisdictions, events, or specific values in transactions.

The Exemption Framework functionality provides a centralised way to handle

jurisdictional exemptions and special conditions identified in different requirements for Argentinian taxes.

This functionality allows banks to manage the exceptions related to jurisdictional taxes, and this is a common functionality used across the Stamp, Turnover Collections, and Turnover Perception taxes.

To handle this configuration, the `ARTAXS.EXEMPTION.PARAMETER` application is introduced to provide the logic to store, read, and apply the settings of the exemptions to be considered for the tax calculation.

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

Taxes » Income Tax on UVA Deposits

This functionality allows users to calculate the income tax with a discount on the maturity date. The income tax is not calculated when the early redeem deposit is done before the maturity date.

The values of the *Monotributo* and *Imp Ganancias* fields, in the AFIP Contributor's file, are taken into account on the income tax calculation.

A customer can have records in the AFIP Contributor's padron, RG2618, and RG830 or not. If the customer has the value of the *Monotributo* field different than NI, this person is considered fully exempted, no matter the information that comes in the *Imp Ganancias* field, or the other padrons.

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

Taxes » Stamp Tax

Stamp tax is a jurisdictional tax managed by the tax authority in each jurisdiction. Each jurisdiction has its own exceptions and rates.

This functionality allows banks to configure the Stamp tax for different products and for different events.

The Stamp tax is supported for the following products: non-instrumented loans, savings accounts, and deposits:

- For non-instrumented loans, the Stamp tax is applied on the initial disbursement and on repayments.
- For savings accounts, the Stamp tax is applied on the outgoing transfers for certain jurisdictions and on the account opening in another jurisdiction.
- For deposits, the Stamp tax is applied on the deposit funding.

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

Taxes » Turnover Collections

The Turnover tax is a jurisdictional tax charged to any person (physical or juridical) that performs an economic activity (including freelancers and companies) and its fiscal residence is any province in Argentina.

For Turnover tax in the collection, the regime is applicable based on the information provided by padrones and not on the customer address itself. Also, exceptions can be added by jurisdiction. For the SIRCREB file, there can be exemptions based on the jurisdiction code.

This functionality allows banks to calculate more than one Turnover tax collection for the same tax event, based on the information coming in padrons. Also, Temenos Transact calculates and charges the Turnover tax for the collection regime for customers, adding as many charges as jurisdictions the

customer is subject to.

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

Taxes » Turnover Perceptions

Turnover tax is a jurisdictional tax charged to any individual person that resides in Argentina and performs an economic activity. The turnover perception is applied on assets.

This functionality allows banks to calculate the turnover perception based on jurisdiction.

The following new fields are added to the `CUSTOMER` application to be used in setting the condition priority:

- *To Perception.*
- *To Perception Jurisdiction Code.*
- *To Perception Rate.*
- *Contrib Afip.*
- *Imp Vat.*
- *Monotax.*

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

Finland Model Bank

Collateral » Collateral Risk View Enquiry

This functionality allows banks to view the risk for a single or group of customers. Collaterals and the limit information of customers are viewed in a single enquiry screen.

The following items are introduced as part of this functionality:

- The `FICOLL.ADDI.LIMIT.DETAIL` application is used to capture the new loan's information given by the user in the Origination system.
- The FIMB.CUSTOMER.SCV.CHECK,FI enquiry is used to display the risk view for a single or group of customers.

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

Collateral » LTV Calculation

This functionality allows banks to calculate the Loan-to-Value (LTV) ratio using the formula provided by the Finnish Supervisory Authority (FSA). Also, this functionality allows users to capture LTV breach reasons applied to housing loans when the Loan-to-Collateral (LTC) ratio is above the allowed value.

The following items are introduced as part of this functionality:

- The `FICOLL.LTV.CALCULATOR` application is used to calculate the LTV numerator and denominator value according to the formula given by FSA whose various parameters are fed in this application.
- The `LIMIT,FSA.LTV` version is used to calculate LTV using the FSA formula.
- The *LTV Denominator* field is added to the `COLLATERAL` application to capture the denominator of the LTV formula given by FSA.

- New fields were added to the FICOLL.COLLATERAL.PARAM application to hold the housing loans details.

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

Collateral » Warning on Amendments Increasing Risk

This functionality allows banks to receive a warning on any amendment that increases the risk for the bank such as releasing any collateral. The bank can decide whether it wants to redo the valuation.

The following items are introduced as part of this functionality:

- The *Nv Lower Breach Cap* and *Nv Upper Breach Cap* fields were added to the FICOLL.COLLATERAL.PARAM application to capture the breach conditions beyond which the warning message has to be displayed.
- The FICOLL.RAISE.COLLATERAL.OVERRIDE version routine is added to validate the difference in the percentage change of the nominal value of the collateral. If the percentage change breaches the allowed limit, then an override message will be displayed by the system.

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

India Model Bank

Lending Compliance » IDPMS

IDPMS (Import Data Processing and Monitoring System) is an Reserve Bank of India (RBI) system introduced to ensure the better monitoring of import of goods and software and facilitating the AD (authorised dealer) banks to report various returns through a single platform. It was launched to overcome the shortcomings in the import reporting processes.

The functionality allows banks to manage the import process and transactions. The system allows the issuance of a bill of entry (BOE), the BOE acknowledgement letter, and if needed, the extension of the BOE, settlement and closure. The functionality also allows the user to upload of an outward remittance message (ORM).

The following applications are introduced as part of the functionality:

- The `INLEND.IMPORT.EXPORT.ATTRIBUTES` application stores all important configuration data and other data for the IDPMS and EDPMS functionalities.
- The `INLEND.IDPMS.ORM` application stores the data for all the outward remittances made.
- The `INLEND.IDPMS.BOE` application stores the data about the bill of entry.
- The `INLEND.BOE.SETTLEMENT` application is used to map the invoices amounts in the BOE with the ORM.
- The `INLEND.IDPMS.BTT` application provides details of the utilised and unutilised amount of an invoice.
- The `INLEND.PORT.LIST` application stores the details of ports designated for imports or Exports and their allotted port codes. The port code is a 6 digit alphanumeric code, allotted by the Government of India.
- The `INLEND.IDPMS.OBB` application is introduced to store the details of the bill of entry tagged to the other bank's AD code and requested for settlement by a different bank.

- Multiple enquiries are provided as part of the functionality to list the details of the IDPMS BOE, BOE settlement, ORM, ORM settlement, and invoice records.

The topic related to this feature is given below:

Lending Compliance

Italy Model Bank

SEDA Creditor Registry » Adoption of eMandates

This functionality allows banks to store the SEDA (SEPA - Compliant electronic database alignment) creditor directory file in Temenos Transact that contains the details of each creditor. The activity includes new, amend and delete. This directory will be published every day by the Italian Banking Association (ABI). This file contains the details of each creditor, the SEDA service to which they have subscribed, their activation date, their creditor bank, if the creditor Id is waived or not from remuneration, etc.

The incoming SEDA directory file will be received in a Temenos Transact folder from where a service will pick up the file and map the same to the `DD.CREDITOR.REGISTRY` (Creditor Registry) application. The process for upload will be done as part of this solution.

> **The topic related to this feature is given below:**
>
> SEDA Creditor Registry

Spain Model Bank

All Fund Bank Interface » API Online Message Generation

When an order is made in Temenos Transact, the order, together with details like the *Distributor Code Api, Sub Distributor Code Api* and the *Product Code* have to be sent to AFB via an API.

This functionality allows banks to send the *Distributor Code Api, Sub Distributor Code Api* and the *Product Code* details in an API message to AFB based on the configuration done in the `ALLDND.AFB.PARAMETER` application.

The following fields are added to the `ALLDND.AFB.PARAMETER` application *Distributor Code Api, Subdistributor Code Api, Application Field, Product Code* and *Application Value*.

> **The topic related to this feature is given below:**
>
> All Fund Bank Interface

All Fund Bank Interface » Funds Catalogue

This functionality allows banks to store the details from the 73 file record provided by Allfunds Bank.

The 73 record type structure has been modified, and the same changes are reflected in the `ALLFND.TRANSACTION.LOG` and `ALLFND.FUND.SECURITY` applications so that Temenos Transact will be able to store the information coming from the 73 record.

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

All Fund Bank Interface

Sri Lanka Model Bank

Foreign Exchange Transactions Reporting » Form 1 and 2 Extract

The functionality allows banks to report the list of foreign exchange transactions carried out by customers to the foreign exchange department in a pre-defined format given by the bank. The report can be extracted as per the frequency configured by the bank. Time schedule can be either every one hour or every two hours. Temenos Transact will extract the data in .txt format delimited with the ~ symbol, and the bank can upload the data into the website of foreign exchange.

The following applications and enquiries are introduced as part of this functionality.

- The `LKFXTR.PARAMETER` application defines the various parameters required for the foreign exchange reporting purchase or sales transactions.
- The `LKFXTR.TXN.DETAILS` application stores the statement entry IDs of the foreign exchange transactions.
- The `LKFXTR.TXN.FILE` application stores the transaction references executed for the day from the configured modules in the `LKFXTR.PARAMETER` application.
- The LKFXTR.REPORT.ERR.DETAILS enquiry is used to list the error details for a particular extraction.

The list of foreign exchange transactions are identified based on the following criteria and will be loaded into the `LKFXTR.TXN.FILE` application. Based on the time schedule, the transactions will be downloaded from this application using a Temenos Transact service and the text file will be generated.

> **The topic related to this feature is given below:**
>
> Foreign Exchange Transactions Reporting

Tunisia Model Bank

Foreign Currency Operations » Business Travel Allowance

This functionality allows users to open, amend and renew business travel allowances (AVA) files. It also allows the manual renewal, suspension and closure of the AVA records.

The following applications and enquiries are introduced with this functionality:

- The `TNFCOP.AVA.ALLOWANCE` application is introduced, which has several versions to perform various activities such as opening an AVA record, amendment, use, manual renewal, close suspension and supply.
- The `TNFCOP.FOREX.PARAM` application is introduced to define the ceilings and percentages for the respective business type. The `TNFCOP.FOREX.PARAM` application facilitates the calculation of the *Eligible Limit Amount*. This is a prerequisite setup before creating an AVA record.
- The `TNFCOP.AVA.ACTIVITY.CODE` application provides a dropdown which will determine the activity codes to the repository of Tunisian activities.
- The TNFCOP.BRANCH.CHECKER enquiry lists the records with the application status as INAU and the *Record Status* as blank in the AVA opening version.
- The TNFCOP.REGULATORY.MAKER.OPENING enquiry will display only those authorised records from the branch which are related to the opening.
- The TNFCOP.REGULATORY.MAKER.AMENDMENT enquiry will display only those authorised records from the branch which are related to the amendment.
- The TNFCOP.REGULATORY.MAKER.RENEWAL enquiry will display only those authorised records from the branch which are related to the renewal.

- The TNFCOP.REGULATORY.CHECKER enquiry lists the records with the application status as INAU, and the *Record Status* as Awaiting Final Regulatory Approval.
- The TNFCOP.BRANCH.MAKER.OPENING.REJECT enquiry lists all the opening records that have been rejected by the regulatory checker.
- The TNFCOP.BRANCH.MAKER.AMENDMENT.REJECT enquiry lists all the amendment records that have been rejected by the regulatory checker.
- The TNFCOP.BRANCH.MAKER.RENEWAL.REJECT enquiry lists all the renewal records that have been rejected by the regulatory checker.
- The TNFCOP.AVA.ALLOWANCE.DETAILS enquiry lists the authorised AVA details.
- The TNFCOP.AVA.USE.SUPPLY.UNAUTH enquiry is used to authorise or delete the INAU records for the usage and supply.
- The TNFCOP.AVA.CLOSURE.SUSP.UNAUTH enquiry is used to authorise or delete the INAU records for suspension and closure.

New fields are introduced in the `TNFCOP.AVA.ALLOWANCE` application that hold information about the charges.

New versions of the `TNFCOP.AVA.ALLOWANCE` application are introduced to hold details of the branch users for opening, amendment and manual renewal.

The beneficiary can be configured using the `RELATION` and `CUSTOMER` applications, and the *Relation* is parameterised in the `TNFCOP.FOREX.PARAM` application.

AVA activities, including usage and supply can be reversed.

> **The topic related to this feature is given below:**
>
> Foreign Currency Operations

Foreign Currency Operations » Foreign Trade Title Clearance - TCE Clearance

This functionality allows banks to send reports to the Central Bank of Tunisia

regarding foreign trade titles. It also allows the update of the imputation, settlement and reservations manually.

The following applications and enquiries are introduces with this functionality:

- The TNFCOP.FOREIGN.TRADE.TITLE application is introduced to receive and store the data from the TTN (Tunisie Trade Net) interface.
- The TNBASE.FINANCIAL.DETAILS application is introduced to receive and store the data received from other applications, and through the manual update.
- The TNFCOP.FOREX.PARAMETER application is introduced to store the default values which are required for trade title operations and for generating the Central Bank report.
- The TNFCOP.TRADE.TITLE.REPORT application is introduced to store the report details which are required to generate the Central Bank report.
- The TNFCOP.TRADE.TITLE.NOTCELARED application is introduced to store the trade titles which are not created during the clearance process, such as during the month end, while updating the CBT (Central Bank of Tunisia) indicator.
- The TNFCOP.AMEND.MANUAL.UPDATE enquiry is introduced to fetch the details of the foreign trade title details, received from TTN and domiciled.
- The TNFCOP.AMEND.MANUAL.SETTLEMENT enquiry is created to fetch the details of the foreign trade title details, which are all not cleared, received from TTN and domiciled.
- The TNFCOP.AMEND.MANUAL.RESERVE enquiry is created to fetch the details of the foreign trade title details, which are all not cleared, that are received from TTN and domiciled.
- The TNFCOP.STATUS.TRADE.TITLE enquiry is created to fetch the trade titles based on the approved reporting status.
- The TNFCOP.CLEARANCE.STATUS enquiry is built to fetch the details of the trade titles which are cleared or not cleared as on date based on the clearance month.

The topic related to this feature is given below:

Foreign Currency Operations

Foreign Currency Operations » Schooling Professional Training File

In Tunisia, there is a practice to produce the schooling professional training file which will be used to make the payments for expenses, fees and other expenses for the schooling or training in foreign countries. The schooling professional training file has to be reported to the Central Bank every month.

This functionality allows banks to manage the schooling professional training file (capture, amend, renew, release, close and export).

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

- The `TNFCOP.FOREX.PARAM, SCHOOL.PROF.INPUT` version is used to set the maximum allowed limit for the monthly expenses and other expenses.
- The `TNFCOP.SCHOOL.PROF.FILE` application is used to create the schooling professional training file. New versions are release to amend, close, renew and release the suspension of the schooling professional training file.
- The `TNBASE.FINANCIAL.DETAILS` application is used to store the used details of the schooling professional training file requested by the customer.
- The TNFCOP.SCHOOL.PROF.FILE.DETAILS enquiry is created to view the schooling professional training file which is created by the branch and which has to be reviewed by regulatory control using the `TNFCOP.SCHOOL.PROF.FILE,REGULATORY` version to be approved or rejected.

- The TNFCOP.FILE.REPORT enquiry is used to fetch the status of the schooling professional training file and prepare an excel or PDF file which has to be shared with the Central Bank.
- The TNFCOP.FILE.STATUS enquiry is used to view the details of the schooling professional training file.
- The TNBASE.SCHOOLING.USAGE enquiry is used to view the details of the usage of the schooling professional training file.

The topic related to this feature is given below:

Foreign Currency Operations

Retail

Arrangement Architecture » Suspend or Resume Accrual or Amortisation of Credit Charge

As per the latest FASB GAAP regulation, the amortisation of a loan origination related credit charge needs to be suspended when the loan moves to non-performing status. AA framework supports the suspend charges functionality for a debit type charge and it maintains the suspended balance separately for accounting purpose. This functionality is extended for credit type charges as well.

It is possible to suspend or resume the accrual or amortisation of a credit type charge for a non-performing loan. When the loan moves to non-performing status, the accrual or amortisation of the underlying credit charge is suspended and the associated accounting entries are posted to suspended balance. When the loan is resumed to performing status, Transact accounts the suspended balance under the respective expense account.

The topics related to this feature are given below:

Suspend or Resume Accrual or Amortisation of Credit Charge - Introduction

Suspend or Resume Accrual or Amortisation of Credit Charge - Configuration

Suspend or Resume Accrual or Amortisation of Credit Charge – Working With

Arrangement Architecture » Stop Suspend and Resume Charge Amortisation

Users can configure charges for loan products and in case the associated arrangement moves into non-performing status, as per the latest FASB GAAP regulation, amortisation of charge need to be stopped instead of suspending. To support this functionality, the Stop Suspend Property Type is introduced in the Charge Property Class. When the loan resumes back to performing status, the amortisation is also resumed and the un-amortized charge balance is booked to P/L.

The Stop Suspend Property Type allows to:

- Stop the amortisation of the charges - both Customer and Non Customer facing charges [ACC<CHARGE>]
- Stop the amortisation of the charge under a suspended loan and it is neither posted to Suspend bucket nor P/L.
- Let the unamortised portion of the charge stays in the same balance until the loan is regularized or resumed.
- Set the charge to *Suspend* or *Stop Suspend*.
- Set charges to Amort or Sets Amort Defer for ‘Stop Suspend’ behaviour.

The topics related to this feature are given below:

Stop Suspend and Resume - Introduction

Stop Suspend and Resume - Configuration

Stop Suspend and Resume – Working with

Arrangement Architecture » Mandatory Alerts for All Delivery Customers

Temenos Transact allows alerts to be sent to other account holders along with

the primary customer for mandatory subscriptions in their preferred channel.

To enable this feature, the *Send Alerts To All* field is introduced in `TEC.ITEMS` that allows the user to define if the mandatory alerts must be sent to all the account holders.

> **NOTE:** This field is available for input only when *Subscription level* is set as Mandatory.

> **The topics related to this feature are given below:**
>
> Mandatory Alerts - Configuration
>
> Mandatory Alerts – Working with

Loans and Deposits » Lending Deposits - Risk-Free Rates

As the IBOR rates are retiring at the end of 2021, to replace these rates, Risk-Free Rates (RFR) are introduced with spread inclusive and exclusive methods in LD module. Risk Free Rates are backward-looking rates and are based on the actual transaction data.

These are published with a lag. Also, no credit risk elements are involved while calculating the risk-free rates.

> **The topics related to this feature are given below:**
>
> Introduction to Processing of Risk-Free Rates
>
> Configuring Risk Free Rates Contracts
>
> Creating Risk Free Rate Contracts
>
> Interest Accruals for RFR Contracts

Technology

Design Framework

Temenos Workbench V2 (Quantum/Cloud) » Creating API Version and API Enquiry

Temenos workbench allows you to create new API Enquiry and API Version by providing the basic information such as application name, product name, API name, API version, operation name, input validation and so on. On successful creation, they are launched in editor (edit mode).

The topics related to this feature are given below:

Creating an API Enquiry

Creating an API Version

Integration Framework

Data Event Streaming » Archive Staging Table

The event cleanup is responsible for redelivering the error events based on the configurations. Since, the DES maintains the staging database (DB), the size of DB grows over a period.

To overcome the size issue in DB, the Staging Table archival strategy is available. This strategy is used to archive the `TMN_DES_EVENT` staging table based on retention days specified by you. An API is also available to perform the archival process.

In addition to event cleanup's functionality, it can archive a record by marking it

as ‘ready for deletion’ based on the EOT message received.

> **The topic related to this feature is given below:**
>
> Archive Staging Table

Integration Framework Runtime (IF) » Redirect Processed Events to a Separate Table

The post-delivery action of integration service allows the `IF.EVENTS.INTERFACE.TABLE` records to either delete or move to a separate file. If data lifecycle module is installed, then the records are not deleted or moved to separate file but the respective service needs to execute to purge the record.

To delete or move the records to a separate file, the ARCHIVE.TO.FILE option is added to the *POST.DELIVERY.ACT* field of `IF.INTEGRATION.SERVICE.PARAM` table. This option in post-delivery action is used to run the purge service and ensure the `IF.EVENTS.INTERFACE.TABLE` is not loaded with processed records.

When you select ARCHIVE.TO.FILE option, it:

- Writes the event record to a *MAIN.FILE* field value of the `IF.EVENTS.INTERFACE.TABLE` record from `ARCHIVE` table. It also deletes the event record from `IF.EVENTS.INTERFACE.TABLE`.
- Allows the user to configure *MAIN.FILE* field value of the `IF.EVENTS.INTERFACE.TABLE` record from `ARCHIVE` table, where you can specify the table name according to the user’s requirement and purge the data to read only DB when required.

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

Redirect Processed Events to a Separate Table

Integration Framework Runtime (ID) » Enriching Integration Events with Temenos Subroutine or API Results

Externalised APIs supports with any number of parameters and internal APIs with one input and one output parameter. This restricts you to send multiple parameters to the API and to get a structured output.

To get a structured output, the `IF.COMPLEX.TYPE` table is now available to handle complex type structure for the fields in the case of internal APIs and allows structured output. Also, it allows Integration Events enriched with values returned from a Temenos Transact subroutine (API) call. The benefits of the new functionality are listed below:

- You can define the field type of complex type.
- You can define any number of input and output parameters for internal APIs with the limit of 15 parameters.
- `IF.FLOW.API` accepts the valid subroutine as a record ID.

  > **NOTE:** Before creating an entry in `IF.FLOW.API`, you should create a subroutine entry in both `PGM.FILE` and `EB.API`.

- You can define the parameters, if provided with the valid subroutine as a record ID in `IF.FLOW.API`.

  > **NOTE:** Parameter must be in the same order as defined in the subroutine.

- If the record ID is SYSTEM in `IF.FLOW.API`, it accepts only one in and one out parameter. If API name is provided in `IF.FLOW.API` with SYSTEM as record ID, then the in and out parameter appears on UI, with the name `<SubroutineName>:”inputParameter”` and `<SubroutineName>:”outputParameter”`.

> **The topic related to this feature is given below:**
>
> Enriching Integration Events with Temenos Subroutine or API Results

Endpoint Connect

Endpoint Connect is a Framework to Consume Integration Messages and call external API's from Enterprise Service Bus (ESB).

It provides an option to consume the input request using a camel-JMS endpoint and sent to processor through camel-processor. The response from the processor is sent as a request through camel-http (client provides the HTTP URL) and the HTTP response is sent to camel-Transformer available at the client end. The final response from the transformer is sent to JMS output queue.

It is a standalone component, which acts as an endpoint to send and receive messages.

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

Interaction Framework

UXP Browser » Custom Enquiry Drilldown Injection

This feature provides lightweight, client-side (JavaScript or jQuery) framework allowing banks to declare custom buttons for automatic injection into targeted Enquiry screens. This feature does not require defining anything new in

Temenos Transact or modifying any models, and so on.

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

IRIS R18 » Single Queue Model

This feature reduces multi-queue dependency and enables IRIS R18 to send all requests and receive responses through single request/response queue. This feature maintains backward compatibility and is completely configurable. To support this feature and distinguish between multi-queue and single-queue requests, the `multiQueueRequest` property is introduced in the `standalone-comms.properties` and `jms.properties` files.

> **The topic related to this feature is given below:**
>
> Single Queue Model

Installation and Configuration Notes

Analytics

Semantic Query Layer » Data Enrichment in Semantic Query Layer

To enable the enrichment rules for Semantic Query Layer module, the following configurations are required.

Application Properties

Cache mechanism type must be set for the framework as `sql.config.cache.type=EHCACHE` in `/SEMANTICQUERYLAYER-X.X.X.ZIP/CONFIG/APPLICATION.PROPERTIES`.

Configuration Properties

Below configuration properties must be available in `/SEMANTICQUERYLAYER-X.X.X.ZIP/CONFIG/ENV/PROD/CONFIG.PROPERTIES`:

- If Metadata source is set to ATLAS, then the below properties should be configured as follows:
  - Meta=data source type property
    `sql.config.metadata.source=ATLAS`
  - TemenosMetaService server details host address, username and password
    `sql.config.atlas.host=<hostaddress>`
    `sql.config.atlas.username=<username>`
    `sql.config.atlas.password=<password>`
- If Metadata source is set to LOCAL, then the below properties should be configured as follows:
  - Metadata source type property
    `sql.config.metadata.source=LOCAL`
  - Metadata inventory files default location path
    `sql.config.metadata.location=../inventory`

Inventory Mapping

The inventory mapping properties are mainly required for LOCAL mode of metadata source usage in Semantic framework. The mapping should be defined between the API operation name and API inventory file name. The mapping provided for pre-released APIs is shown below:

```
getAllReportTimes_v100@Api=getServiceDetails-v1.0.0-inventory
getAllReportTimesCOB_v100@Api=getServiceDetails-v1.0.0-inventory
getProductLineActivities_v100@Api=getProductDetails-v1.0.0-inventory
getProductLines_v100@Api=getProductDetails-v1.0.0-inventory
getLanguageList_v100@Api=getLanguageList-v1.0.0-inventory
getSectorList_v100@Api=getSectorList-v1.0.0-inventory
getRateTexts_v100@Api=getRateTexts-v1.0.0-inventory
```

When any new API is created, mapping should be specified between API operation name and API inventory filename in `/SEMANTICQUERYLAYER-X.X.X.ZIP/CONFIG/INVENTORY-MAPPING.PROPERTIES` property file.

Source Mapping

The source mapping property is required to specify the mapping between Temenos Transact source application names and ODS table name. The mapping provided for pre-releases APIs is shown below:

```
app.map.AA_ACTIVITY_CLASS=AA.ACTIVITY.CLASS
app.map.AA_ACTIVITY_CLASS_ACTION=AA.ACTIVITY.CLASS
app.map.AA_ACTIVITY_CLASS_ACTIVITY_TYPE=AA.ACTIVITY.CLASS
app.map.AA_ACTIVITY_CLASS_ALERT_EVENT_TYPE=AA.ACTIVITY.CLASS
app.map.AA_ACTIVITY_CLASS_ATTRIBUTE=AA.ACTIVITY.CLASS
app.map.AA_ACTIVITY_CLASS_BATCH_NAME=AA.ACTIVITY.CLASS
```

When a new API is created, mapping for the tables involved in the API should be defined in `/SEMANTICQUERYLAYER-X.X.X.ZIP/CONFIG/SRC-MAPPING.PROPERTIES` property file.

Banking Framework

System Tables » Generating Accrual and Amortisation Entries through Soft Accounting

The following records are released to support the soft accounting call from the EB.ACCRUAL application for AA Arrangement contracts.

- AC.EVENT
  - CHARGE-ACCRUAL-ADJ
  - CHARGE-ACCRUAL-AMORTISE
  - CHARGE-PL-REVESE.TO.SUS
  - CHARGE-ACCRUAL-RESUME
  - CHARGE-ACCRUAL-REVERSE
  - CHARGE-ACCRUAL-SUS-REVERSE
- AC.POSTING.DETAIL
  - CATEG-ACCR
  - SPEC-ACCR
  - STMT-ACCR
- AC.ALLOCATIN.RULE
  - CHARGE-ACCRUAL

Extensibility APIs

Java Extensibility

Category: New Enhanced Existing Deprecated

Package	Class	Method name	Description	Hook*/ API
payments	PaymentLifecycle	**getFileName**	Enables the developer to get a file name for the outgoing payment message type.	Hook
system	Record	**getGroupName**	Returns a group name for the given record based on the conditions specified for the given table name.	API
system	Enquiry	**setDropdownFilterCriteria**	Sets the selection criteria using the values of the current record from an input screen.	Hook
system	DataAccess	**getCurrentDirectory**	Returns the Temenos Transact current runtime directory path.	API
contract	ProvisionManagement	**getSegmentationProvision**	Enables the developer to get segmentation provision percentages to be used in provision calculation.	Hook

Package	Class	Method name	Description	Hook*/ API
*Hooks are placeholders in Transact where routines can be attached to an application. For example, version, enquiry, delivery and so on.				