

fintech 

Distribution Management 8.0 CTP User Guide

TOC

Overview	4
Business Logic	4
Availability & Main Features	5
Product Factory	6
Subledger	6
Loan Management	7
Initial Configurations	8
Agreement Types	9
Creating Agreement Types	9
Viewing Agreements of a Specific Agreement Type	13
Invoice Types	13
Creating Invoice Types	14
Invoicing Configuration	15
Creating Invoicing Configurations	15
Transaction Types	27
Predefined Transaction Types	28
Creating Transaction Types	28
System Parameters	29
Managing Distribution Management System Parameters	33
External Request Processing	35
Configuring the Async Engine	35
Managing Unprocessed Requests	43
Scheduled Jobs	46
Security Roles	47

Predefined Distribution Management Security Roles	48
Commission	54
Creating Third-Party Commissions	56
Commission Schema	60
Creating Third-Party Commission Schemas	61
Commission Type	61
Creating Third-Party Commission Types	62
Managing Third-Party Entities	63
Creating Third-Party Entities & Assigning Roles	64
Managing Third-Party Agreements	66
Third-Party Agreements Life Cycle and States	67
Third-Party Agreements Versioning	68
Changing Third-Party Agreement Statuses	68
Creating Agreements For Third-Parties	70
Approving a Third-Party Agreement	77
Rejecting a Third-Party Agreement	79
Creating New Versions Of Third-Party Agreements	80
Viewing a Third-Party Agreement's History	80
Working with Third-Party Invoices	82
Managing Invoices For Third-Parties	83
Third-Party Invoices Life Cycle and States	84
Changing Third-Party Invoice Statuses	85
Creating Third-Party Invoices	85
Managing Automatic Invoice Payments	94
Dashboards and Reports	99
Third-Party Agreements	99

Overview

FintechOS **Distribution Management** helps you manage third-party entities (agents, brokers, insurers, etc.) registered with your institution to intermediate the selling of various products to customers. For their work, the third-party entities are compensated with fees payable for each new sale, based on a pricing agreement with your institution. Distribution Management facilitates the creation and management of third-party agreements based on approvals, the linkage of contracts/ deals to third-parties, and the configuration of the commissioning processes through dedicated menus. A third-party invoicing process also takes care of the transfer part of the payments related with these third-party entities.

All the features in Distribution Management are built using the capabilities of **FintechOS Studio**, and you can access its menus and dashboards when logged in **FintechOS Portal**.

Business Logic

Let's say a third-party entity (an agent, a broker, an insurer, or a merchant) agrees with an institution to intermediate the selling of various products to customers, for a fee. Thus, an agreement is recorded in Distribution Management, containing all the pricing information needed to compensate the third-party. Specific commissioning configurations must be in place to be then applied to the agreements. Whenever the third-party entity intermediates the selling of a product to a customer, specifying in the contract the entity and their role, the entity should be compensated with the commissions mentioned in the agreement. The payments are performed based on automatic or manual invoicing processes. Agreement records must be approved before being used for invoice generation. You can create invoices and attach invoice details for an agreement for each currency mentioned in the pricing details, that are automatically processed for payment by Distribution Management. An automated process running once each night creates third-party invoices and payments for the combination of third-party entity/ agreements currency, during the validity of the

agreement, on the Payment Day of each agreement, excluding any invoice details already created on a manual invoice. The payments for the invoices are performed for unallocated or partially allocated payments.

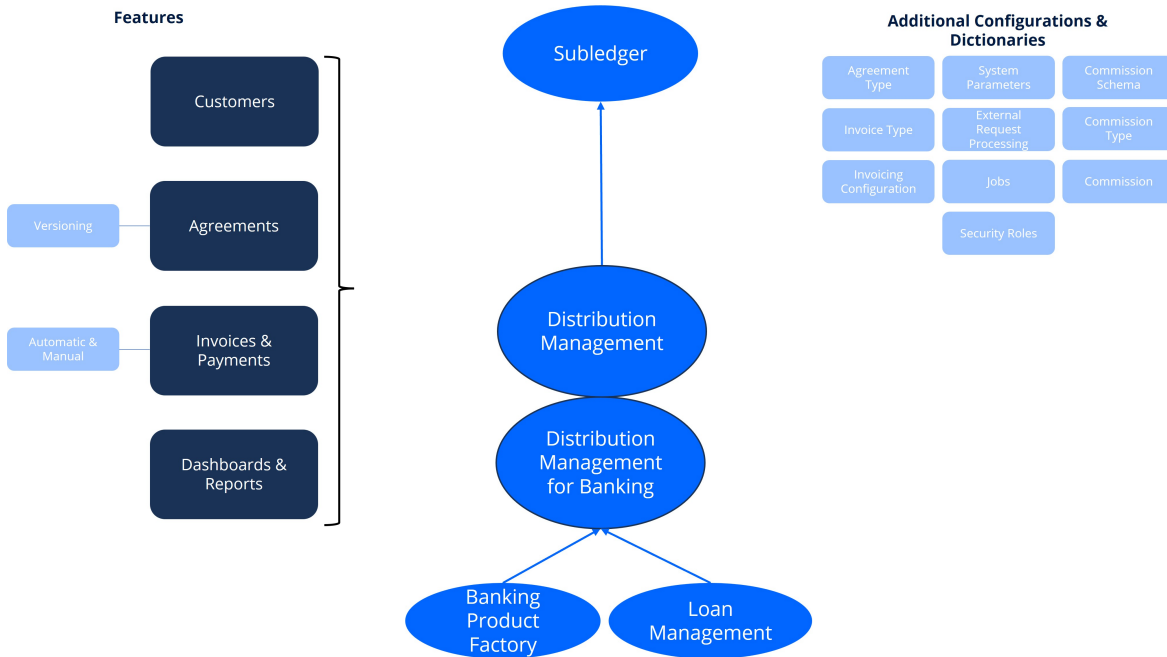
Availability & Main Features

Distribution Management is available either as a stand-alone solution, decoupled from any specific business, or as an add-on for banking-specific business, installed on top of the FintechOS Loan Management package.

NOTE

This user guide focuses on the features available with the stand-alone Distribution Management package. Any banking or insurance-specific functionality is clearly specified in the guide's pages.

The diagram below exhibits the main features of Distribution Management, along with a series of configurations and dictionaries used to automate the complex processes performed by the system. Distribution Management uses the banking products defined within **Banking Product Factory**. Distribution Management records can be further used by **Subledger** to generate ledger entries.



Product Factory

This is a powerful automation processor accessible in the FintechOS Studio that builds the products to be used in a customer journey, configures the interest, commissions and the life cycle of a product. Those products are later introduced into a [customer journey](#) or, when used in conjunction with Loan Management, they associate the products with [transaction types](#). For more information, see [Product Factory](#).

Subledger

Based on the transactions performed in Distribution Management, Subledger logs, along with a company's financial transactions, details that enable the system to build ledger entries. For more information, see [Subledger](#).

Loan Management

Loan Management, accessible in the FintechOS Portal, controls the major banking processes and configurations that support your banking products, such as limits, collateral, groups, or contracts. For more information, see [Loan Management](#).

Initial Configurations

This page contains a series of topics that explain the configurations needed by Distribution Management, either during implementation, or when you need to adjust the commissions used for third-party entities payments. The user guide follows the logical steps for configuring Distribution Management. Follow these steps before starting to work with agreements and invoices.

IMPORTANT!

FintechOS offers you a [Utilities package](#) for banking solutions, which you can use for testing purposes on your pre-production environments. This package helps you test the FintechOS banking solutions during the implementation phase of your project, using sample data. As such, you'd benefit from its content if you are a developer, a tester, a digital consultant, or a partner who works on implementing one of the FintechOS banking solutions.

Check the following configurations during the implementation process:

- [Agreement Types](#)
- [Invoice Types](#)
- [Invoicing Configuration](#)
- [Transaction Types](#)
- [System Parameters](#)
- [External Request Processing](#)
- [Scheduled Jobs](#)
- [Security Roles](#)

The following pages explain the configurations needed by Distribution Management regarding the setup of commissions for third-party entities:

- [Commission Schema](#)
- [Commission Types](#)

- **Commissions**

IMPORTANT!

Your user must have the one of the following security roles to perform these configurations: [Distribution Management Admin](#) (when Distribution Management is implemented) or [Distribution Management Admin Banking](#) (when Distribution Management for Banking is implemented).

Agreement Types

Distribution Management can manage third-party entity agreements of multiple types, according to your needs. Each agreement type has a different configuration that must be specified during the implementation process, containing information about the custom-built product type and product endpoints that retrieve the information needed to create the agreements.

To manage agreement types:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Settings** > **Agreement Types**.
2. On the **Agreement Types** page, you can create a new agreement type record, search, edit, or delete an existing one. You can't delete records already used by other records.

Creating Agreement Types

Follow these steps to create new third-party agreement type records:

1. In the FintechOS Portal, click the **Insert** button. The **Agreement Type** page is displayed.

2. Fill in the following fields from the **Main Info** section:
 - **Name** - Enter the name of the agreement type.
 - **Is Default** - Check this box to specify that this agreement type should be used by default by Distribution Management when creating an agreement.
 - **API Product Type** - Enter the name of the custom-built product type endpoint that retrieves information to [build the agreement](#).
 - **API Product** - Enter the name of the custom-built product endpoint that retrieves information to [build the agreement](#).

IMPORTANT!

When building these custom endpoints during the implementation process, note that both endpoints should get the data in the following format:

```
{"id": "primary_key_of_the_record_such_as_bankingProductId", "name": "name_of_the_record"}
```

The **API Product Type** endpoint filters the results of the **API Product** endpoint. Keep in mind that the **API Product** endpoint must receive the `productType` parameter, with this exact name. Use the `productType` parameter's value to filter the list of products returned by the endpoint.

Examples of Custom-Built Endpoints' Requests & Responses

Consult the examples below to see the requests and responses of custom-built endpoints for integration with Loan Management:

**Product Type endpoint example: FTOS_TPM_
GetBankingProductTypeForPricing**

Request

```
{}
```

Response

```
[
  {
    "id": "d3c33938-5e0f-4f09-9b12-1068a594b903",
    "name": "BNPL",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "99338609-bdfb-4ecc-9aa3-6dfb3efefbdd",
    "name": "Credit Card",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "a64e8e56-8359-42d7-a37b-148320f3fea6",
    "name": "Deposit",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "b7e70213-372e-4b6e-97f9-d16269578d0f",
    "name": "Leasing",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "60b81570-faa0-4dcc-b060-b708538fa3f2",
    "name": "Mortgage",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "1fd781bc-7242-43fb-98ed-0e489782faca",
    "name": "Overdraft",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "1ec33a8b-996b-4cb8-8f05-c8cc239dc844",
    "name": "Credit Line",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "2e328886-f355-4b9f-8b3b-504fc21c6d1c",
    "name": "Current Account",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "16a94f38-a343-444b-b848-7ad3e574877a",
    "name": "Current Account Overdraft",
    "entityName": "FTOS_BP_BankingProductType"},
  {
    "id": "2aeb312f-2388-417e-ac21-99c252d8ea7b",
```

```

    "name": "Debit Card",
    "entityName": "FTOS_BP_BankingProductType"},
    {"id": "e9b1d150-f5d6-4143-86df-4643dd5e101a",
    "name": "Savings",
    "entityName": "FTOS_BP_BankingProductType"},
    {"id": "f5ce5e29-b013-46a2-9ae7-6e1feb0ef026",
    "name": "Secured Loan",
    "entityName": "FTOS_BP_BankingProductType"},
    {"id": "3c063060-a6b9-434a-ae83-da5df5191291",
    "name": "Unsecured Loan",
    "entityName": "FTOS_BP_BankingProductType"},
    {"id": "cf8b1934-8caa-48d8-b920-134456b1f42f",
    "name": "Working Capital",
    "entityName": "FTOS_BP_BankingProductType"}
  ]

```

**Product endpoint example: FTOS_TPM_
GetBankingProductForPricing:**

Request

```

{"productType": ["3c063060-a6b9-434a-ae83-
da5df5191291"]}

```

Response

```

[
  {
    "id": "30411fcb-e7ba-4116-90b9-
e6b6f88372ee",
    "name": "StraightLoan_BU",
    "entityName": "FTOS_BP_BankingProduct"
  }
]

```

3. Click the **Save and Reload** button.

Viewing Agreements of a Specific Agreement Type

To view all the agreements of a specific type generated by Distribution Management, follow these steps:

1. In the FintechOS Portal, click the desired agreement type record within the **Agreement Types** page. Click the **Agreement** tab.
2. View the list of existing agreements. You can view the following information about each agreement within the list: name, validity period, third-party entity name, and agreement status.
3. Double-click an agreement from the list to display the agreement record.

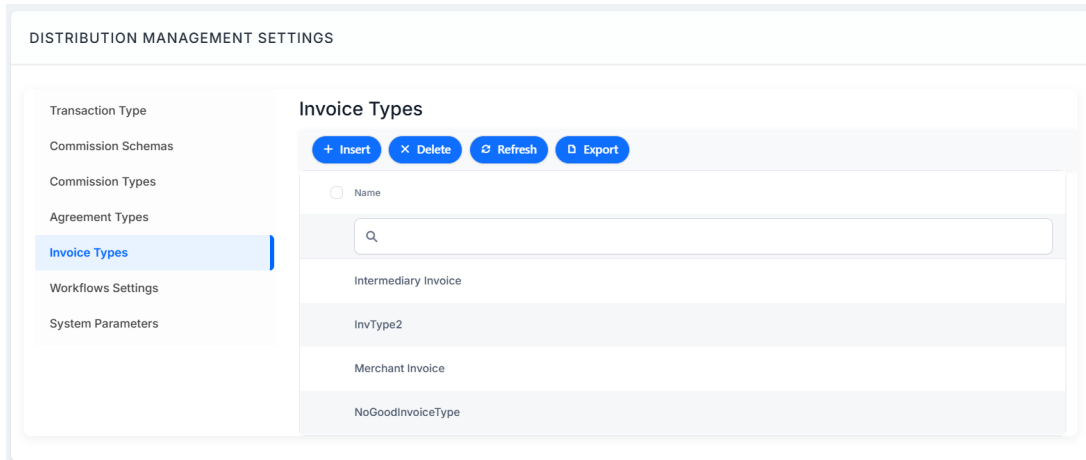
Invoice Types

You can manage the invoice types generated by Distribution Management through the **Invoice Type** dictionary menu item. The invoice types are later used when creating the Distribution Management [invoicing configurations](#).

To manage invoice types:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Settings** > **Invoice Types**.
2. On the **Invoice Types List** page, you can create a new invoice type record, search, edit,

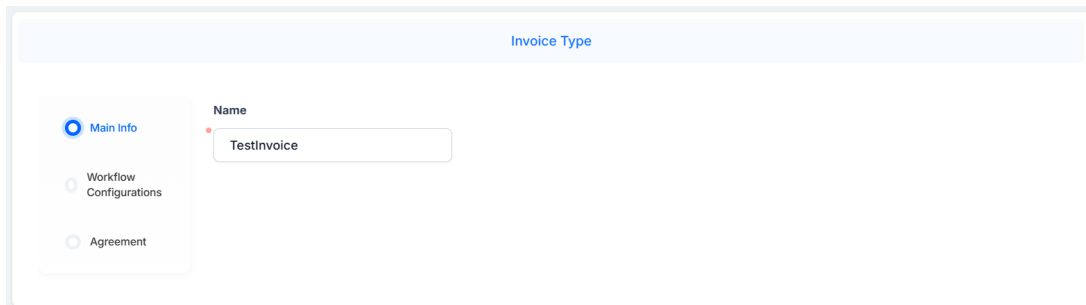
or delete an existing one. You can't delete records already used by other records.



Creating Invoice Types

Follow these steps to create new invoice type records:

1. In the FintechOS Portal, click the **Insert** button. The **Invoice Type** page is displayed.



2. In the **Main Information** section, fill in the **Name** field with the desired name for the invoice type.
3. Click the **Save and Reload** button.

Invoicing Configuration

You can manage the compensation of third-party entities through different invoicing processes, defined per your specific business requirements. For example, you could set up an invoicing process to be used in conjunction with the contracts recorded in Loan Management, or you could use a separate invoicing process for insurance policies. Through the **Workflow Configurations** menu item, Distribution Management allows you to specify which custom-made function to use to retrieve the invoice details for each scope and invoicing type.

To manage third-party invoicing configurations:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Settings** > **Invoice Types**.
2. On the **Invoice Types List** page, you can create a new invoice type record, search, edit, or delete an existing one. You can't delete records already used by other records.

Creating Invoicing Configurations

Follow these steps to create new invoicing configuration records:

1. On the Invoice Type page, click the **Workflow Configurations** tab.
2. Click the **Insert** button to add configurations.
3. Fill in the following fields from the **Main Information** section:

The screenshot shows a 'Workflows Settings' form with the following fields and values:

- Main Info** (tab selected)
- Invoicing Type**: Merchant Invoice
- Scope**: Invoice
- Name**: CB Invoice
- Execute Insert**: No
- Library Name**: FTOS_CB_TPM_InvoiceHelper
- Function Name**: sendDataForCalculation_CB
- Has Extended Attributes**: Yes
- Extended Attributes**: ContractId,EventId

- **Name** - Enter the name of the configuration.
- **Scope** - Select the scope of the configuration. You can select one of the following options: Invoice or Payment.
- **Invoicing Type** - Choose the invoicing type to filter only the agreements of the selected type to be considered through the current configuration. The invoicing type must be set at each agreement's level.
- **Library Name** - Specify the name of the library containing the function(s) you want to use for this configuration scope.
- **Function Name** - Specify the function(s), **custom-built for your business**, from the library that will be executed for the current scope.

IMPORTANT!

Make sure that the function you define for the Invoice scope follows the [input and output example below](#).

For the Payment scope, the function is executed when you change the invoice's business status from NotPaid to Paid. For each setting found in the order of orderIndex attribute, the system imports the library and calls function. If an error is received, then the status is not changed.

If error is received that the status cannot be changed

- **Has Extended Attributes** - Check this box to specify if an invoicing detail has references to other entities. If there are no references, leave it unchecked.
- **Extended Attributes** - If Has Extended Attributes = True, then list here the names of the attributes referenced in other entities. They must exist at the invoice detail level.
- **Execute Insert** - Check this box if the function you just entered also inserts the invoice detail (for invoice scope). If not, the function must have a specific output structure that will be used by Distribution Management to perform the insert. If the function also performs the insert, then the

output structure is not relevant.

4. Click the **Save and Reload** button. The configuration record is saved.

IMPORTANT!

If you are using multiple functions for one scope, you should order the functions on the **Distribution Management Configurations List** page by dragging and dropping them in the correct order (displayed by the **Order Index** attribute) so the system knows which is the correct way of calling them.

Input/ Output Structures for the Function Defined for Invoice Scope

Function input example:

```

Input
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "array",
  "description": "Array of objects containing
agreement/invoice details details.",
  "items": [
    {
      "type": "object",
      "description": "Object containing agreement/invoice
details details.",
      "properties": {
        "agreement": {
          "type": "object",
          "description": "Agreement object.",
          "properties": {
            "agreementId": {
              "type": "string",
              "description": "Id of the agreement."
            },
            "validFrom": {
              "type": "object",
              "description": "Agreement validFrom
property.",
              "properties": {
                "invariantDate": {
                  "type": "string"
                }
              }
            }
          }
        }
      }
    }
  ],
}

```

```

        "required": [
            "invariantDate"
        ]
    },
    "validTo": {
        "type": "object",
        "description": "Agreement validFrom
property.",
        "properties": {
            "invariantDate": {
                "type": "string"
            }
        },
        "required": [
            "invariantDate"
        ]
    },
    "required": [
        "agreementId",
        "validFrom",
        "validTo"
    ],
    "invoiceDetails": {
        "type": "array",
        "description": "Array containing invoice detail
objects.",
        "items": [
            {
                "type": "object",
                "description": "Object representing an invoice
detail.",
                "properties": {
                    "invoiceId": {
                        "type": "null",
                        "description": "The invoice id, should be
null."
                    },
                    "value": {
                        "type": "null",
                        "description": "The calculated value,
should be null, it will be calculated by the next endpoint."
                    },
                    "currencyId": {

```

```

        "type": "string",
        "description": "The currency of the
calculated value."
    },
    "exchangeRate": {
        "type": "number",
        "description": "The exchange rate between
the commission currency and agreement currency."
    },
    "equivalentValue": {
        "type": "null",
        "description": "The calculated
equivalentvalue, should be null, it will be calculated by
the next endpoint."
    },
    "detailTypeId": {
        "type": "string",
        "description": "The detail type id, in
case of a normal commission."
    },
    "clawbackDetailTypeId": {
        "type": "string",
        "description": "The detail type id, in
case of a clawback commission."
    },
    "tpmCommissionId": {
        "type": "string",
        "description": "The commission id."
    },
    "tpmClawbackCommissionId": {
        "type": "string",
        "description": "The clawback commission
id."
    },
    "agreementId": {
        "type": "string",
        "description": "The agreement id."
    },
    "agreementPricingId": {
        "type": "string",
        "description": "The agreement pricing id."
    },
    "calculateCommValueData": {
        "type": "object",

```

```

        "description": "Object containing
calculation data for a normal commission",
        "properties": {
            "percentAppliedTo": {
                "type": "string",
                "description": "The id of the
dimension to which the percent will be applied."
            },
            "valuePercent": {
                "type": "number",
                "description": "The percentage value
that will be applied."
            }
        },
        "required": [
            "percentAppliedTo",
            "valuePercent"
        ]
    },
    "calculateClawbackCommValueData": {
        "type": "object",
        "description": "Object containing
calculation data for a clawback commission",
        "properties": {
            "percentAppliedTo": {
                "type": "string",
                "description": "The id of the
dimension to which the percent will be applied."
            },
            "valuePercent": {
                "type": "number",
                "description": "The percentage value
that will be applied."
            }
        },
        "required": [
            "percentAppliedTo",
            "valuePercent"
        ]
    },
    "clawbackValue": {
        "type": "number",
        "description": "The clawback calculated
value, should be null, it will be calculated by the next
endpoint."
    }

```

```

    },
    "tpmClawbackCommPeriodTypeId": {
      "type": "string",
      "description": "The period type id needed
to calculate if clawback can still be performed."
    },
    "tpmClawbackCommPeriod": {
      "type": "number",
      "description": "The period value."
    },
    "tpmClawbackCommBlockAfterClawback": {
      "type": "boolean",
      "description": "Specifies if clawbacks
will be prevented after the first clawback occurred."
    },
    "clawbackEquivalentValue": {
      "type": "number",
      "description": "The calculated
equivalentvalue for clawback, should be null, it will be
calculated by the next endpoint."
    },
    "clawbackExchangeRate": {
      "type": "number",
      "description": "The exchange rate between
the clawback commission currency and agreement currency."
    }
  },
  "required": [
    "invoiceId",
    "value",
    "currencyId",
    "exchangeRate",
    "equivalentValue",
    "detailTypeId",
    "clawbackDetailTypeId",
    "tpmCommissionId",
    "tpmClawbackCommissionId",
    "agreementId",
    "agreementPricingId",
    "calculateCommValueData",
    "clawbackValue",
    "tpmClawbackCommPeriodTypeId",
    "tpmClawbackCommPeriod",
    "tpmClawbackCommBlockAfterClawback",
    "clawbackEquivalentValue",

```

```

        "clawbackExchangeRate"
      ]
    }
  ]
},
"required": [
  "agreement",
  "invoiceDetails"
]
}
]
}

```

Function Output example:

Output example

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "type": "object",
  "description": "Object containing the details needed for
calculation and insertion.",
  "properties": {
    "extendedAttributes": {
      "type": "object",
      "description": "Object containing extended attributes
if there are any defined for your use-case.",
      "properties": {
        "extendedAttributeId": {
          "type": "string",
          "description": "The extended attribute id. It's
name should be changed to adapt to your use-case and should
be the same as the extended attributes defined in FTOS_TPM_
Configuration for Invoice scope."
        }
      },
      "required": [
        "extendedAttributeId"
      ]
    },
    "invoiceDetails": {
      "type": "array",
      "description": "Array of invoice detail objects.",
      "items": [

```

```

    {
      "type": "object",
      "description": "Invoice detail object.",
      "properties": {
        "invoiceId": {
          "type": "null",
          "description": "Invoice id, should be null."
        },
        "value": {
          "type": "null",
          "description": "The calculated value, should
be null, it will be calculated by the next endpoint."
        },
        "currencyId": {
          "type": "string",
          "description": "The currency of the calculated
value."
        },
        "exchangeRate": {
          "type": "number",
          "description": "The exchange rate between the
commission currency and agreement currency."
        },
        "equivalentValue": {
          "type": "null",
          "description": "The calculated
equivalentvalue, should be null, it will be calculated by
the next endpoint."
        },
        "detailTypeId": {
          "type": "string",
          "description": "The detail type id, in case of
a normal commission."
        },
        "clawbackDetailTypeId": {
          "type": "string",
          "description": "The detail type id, in case of
a clawback commission."
        },
        "tpmCommissionId": {
          "type": "string",
          "description": "The commission id."
        },
        "tpmClawbackCommissionId": {
          "type": "string",

```

```

    "description": "The clawback commission id."
  },
  "agreementId": {
    "type": "string",
    "description": "The agreement id."
  },
  "agreementPricingId": {
    "type": "string",
    "description": "The agreement pricing id."
  },
  "calculateCommValueData": {
    "type": "object",
    "properties": {
      "percentAppliedTo": {
        "type": "string",
        "description": "Object containing
calculation data for a normal commission",
      },
      "valuePercent": {
        "type": "number",
        "description": "The percentage value that
will be applied."
      }
    },
    "required": [
      "percentAppliedTo",
      "valuePercent"
    ]
  },
  "clawbackValue": {
    "type": "number"
  },
  "calculateClawbackCommValueData": {
    "type": "object",
    "description": "Object containing calculation
data for a clawback commission",
    "properties": {
      "percentAppliedTo": {
        "type": "string",
        "description": "The id of the dimension to
which the percent will be applied."
      },
      "valuePercent": {
        "type": "number",

```

```

        "description": "The percentage value that
will be applied."
    },
    },
    "required": [
        "percentAppliedTo",
        "valuePercent"
    ]
},
"tpmClawbackCommPeriodTypeId": {
    "type": "string",
    "description": "The period type id needed to
calculate if clawback can still be performed."
},
"tpmClawbackCommPeriod": {
    "type": "number",
    "description": "The period value."
},
"tpmClawbackCommBlockAfterClawback": {
    "type": "boolean",
    "description": "Specifies if clawbacks will be
prevented after the first clawback occurred."
},
"clawbackEquivalentValue": {
    "type": "number",
    "description": "The calculated equivalentvalue
for clawback, should be null, it will be calculated by the
next endpoint."
},
"clawbackExchangeRate": {
    "type": "number",
    "description": "The exchange rate between the
clawback commission currency and agreement currency."
},
"name": {
    "type": "string",
    "description": "The name of the invoice detail
entry."
},
"clawbackStartDate": {
    "type": "object",
    "description": "The date from which the
clawback period will be calculated.",
    "properties": {
        "invariantDate": {

```

```

        "type": "string"
      }
    },
    "required": [
      "invariantDate"
    ]
  },
  "extendedAttributeId": {
    "type": "string",
    "description": "The extended attribute. Name
should match the extended attribute(s) defined in FTOS_TPM_
Configuration for Invoice scope.",
  },
  "isClawback": {
    "type": "boolean",
    "description": "Specifies if the invoice
detail generated is for a clawback or not."
  }
},
"required": [
  "invoiceId",
  "value",
  "currencyId",
  "exchangeRate",
  "equivalentValue",
  "detailTypeId",
  "clawbackDetailTypeId",
  "tpmCommissionId",
  "tpmClawbackCommissionId",
  "agreementId",
  "agreementPricingId",
  "calculateCommValueData",
  "clawbackValue",
  "tpmClawbackCommPeriodTypeId",
  "tpmClawbackCommPeriod",
  "tpmClawbackCommBlockAfterClawback",
  "clawbackEquivalentValue",
  "clawbackExchangeRate",
  "name",
  "clawbackStartDate",
  "isClawback"
]
}
]
}
},

```

```
"required": [  
  "extendedAttributes",  
  "invoiceDetails"  
]  
}
```

Transaction Types

Any transfer of funds between two bank accounts is recorded as a transaction. There are different types of transactions used in the financial world. You can manage the transaction types in several solutions, depending on which solutions are integrated with your implementation of Distribution Management. For example, if you are using Distribution Management in conjunction with Loan Management, you can use the FintechOS Portal's **Admin Configurations -> Transaction Type** menu. See more details about managing transaction types in the [SubledgerUser Guide](#).

Distribution Management allows you to define transaction types that are later available at the agreement pricing level through a dedicated menu entry. Transaction types defined using the **Distribution Management Settings -> Transaction Type** menu can only be used in Distribution Management processes.

NOTE

For more complex transaction type definitions, we recommend you to use the **Transaction Type** menus available in the Loan Management, Subledger, or Product Factory solutions, if any of these is implemented along with Distribution Management.

To manage transaction types in Distribution Management:

1. In FintechOS Portal, click the main menu icon > **Distribution Management > Distribution Management Settings > Transaction Type**.
2. On the **Transaction Type** page, you can create a new transaction type record, search, edit, or delete an existing one. You can't delete records already used by other records.

Predefined Transaction Types

If you are using Distribution Management without any other FintechOS banking solutions, then there are no predefined transaction types already defined in the system. [Create your transaction types](#) during the implementation process, to be used along with agreement pricing records. Create new transaction types as required by your expanding business, when you add business lines targeted for commissioning.

NOTE

When Distribution Management is used with Loan Management, you can use the [transaction types that are predefined in Loan Management processes](#). Loan Management already contains the following transaction types related to third party entity management processes:

- **Agreement** - It represents a binding contract between an institution and a third-party entity (agent, broker, insurer, etc.) to formalize an agreement to financially compensate the third-party for the inter-mediating the selling of products or services to customers, or compensate the institution for the inter-mediation of selling the third-party's products or services to customers, and to compensate the institution for managing the contract with the third-party.
- **Third-Party Invoice** - It represents the invoice through which the amounts automatically calculated based on an agreement are recorded in Distribution Management.

Creating Transaction Types

Follow these steps to create new transaction type record to be used with Distribution Management:

1. Click the **Insert** button, the **Transaction Type** page is displayed.

The screenshot shows a web form titled "Transaction Type". On the left, there is a tab labeled "Main Info" with a blue circle icon. The form contains three input fields: "Name" with the text "Accruals and Provisions", "Transaction Code" with the text "AP", and "Is Clawback Transaction" which has two radio buttons, "Yes" (which is selected) and "No".

2. Fill in the following fields from the **General Data** section:
 - **Name** - Enter the name of the transaction type.
 - **Transaction Code** - Enter a unique code for the transaction type.
 - **Is Clawback Transaction** - Check this box if the transaction type accepts clawback transactions.
3. Click the **Save and Reload** button.

System Parameters

The system parameters used by Distribution Management determine the behavior of all the Distribution Management features.

NOTE

Do not confuse the Distribution Management system parameters with the FintechOS system parameters, stored in the `systemparameter` and `systemParameterOnPortalProfile` entities!

Predefined Distribution Management System Parameters

Here's the list of system parameters used by Distribution Management, along with their description:

GenerateInvoiceDetailsFlowName

The parameter is used for the external request processing through Async Engine, when triggering a request for the generate invoice details flow.

Module that uses the system parameter: Distribution Management

Parameter type: Text.

Default value: Flow_TPM_GenerateInvoiceDetails_
AsyncEngine

GenerateInvoicesFlowName

The parameter is used for the external request processing through Async Engine, when triggering a request for the generate invoices flow.

Module that uses the system parameter: Distribution Management

Parameter type: Text.

Default value: Flow_TPM_GenerateInvoices_AsyncEngine

PayInvoicesFlowName

The parameter is used for the external request processing through Async Engine, when triggering a request for the invoices payment flow.

Module that uses the system parameter: Distribution Management

Parameter type: Text.

Default value: Flow_TPM_PayInvoices_AsyncEngine

FTOS_TPM_ExchangeRate_UseLatest

It specifies whether Distribution Management should use the latest available exchange rates for calculations (for True value, when you know that exchange rate updates are not performed on a daily basis) or the current date's exchange rate (for False value).

Module that uses the system parameter: Distribution Management

Parameter type: Boolean

Default value: True

PaymentIsNet

It specifies whether Distribution Management should generate one or two bank account transactions and payments for a [third-party agreement invoice](#) when the invoice's status is changed from **Approved to Unpaid**.

For `False` value, two bank account transactions are generated with two payments:

- One transaction with **source** account = Settlement Account and **destination** account = Reconciliation Account with the value of Total Amount To Recover;
- Another transaction with **source** account = Reconciliation Account and **destination** account = Settlement Account with the value of Total Amount To Pay.

For `True` value, Distribution Management calculates the difference between Total Amount To Recover and Total Amount To Pay. Only one bank account transaction is generated and only one payment, representing the non-zero value between the Total Amount To Recover and the Total Amount To Pay, as follows:

- If $\text{Total Amount To Recover} - \text{Total Amount To Pay} > 0$, a new bank account transaction is generated with **source** account = Settlement Account and **destination** account = Reconciliation Account, and a payment is generated for the invoice.
- If $\text{Total Amount To Recover} - \text{Total Amount To Pay} = 0$, a bank account transaction is generated, and the transaction's status changes to **Paid**.

- If `Total Amount To Recover - Total Amount To Pay < 0`, a new bank account transaction is generated with **source** account = `Reconciliation Account` and **destination** account = `Settlement Account`, and a payment is generated for the invoice.

When the payments are approved, the invoice's status becomes **Paid**.

Module that uses the system parameter: Distribution Management

Parameter type: Boolean

Default value: False

ThirdPartyRole

It contains the list of allowed roles to choose from in the [third-party agreement](#) form.

Module that uses the system parameter: Distribution Management

Parameter type: Text

Default value: Merchant, Insurer, Broker, Agent

UseCBTPM

It specifies if the Distribution Management for Banking package is installed on your system. If `UseCBTPM = True`, the system uses Loan Management-specific attributes (such as settlement account and reconciliation account) for invoice payments.

Module that uses the system parameter: Distribution Management

Parameter type: Boolean

Default value: False. It is automatically updated to True when the Distribution Management for Banking package is installed.

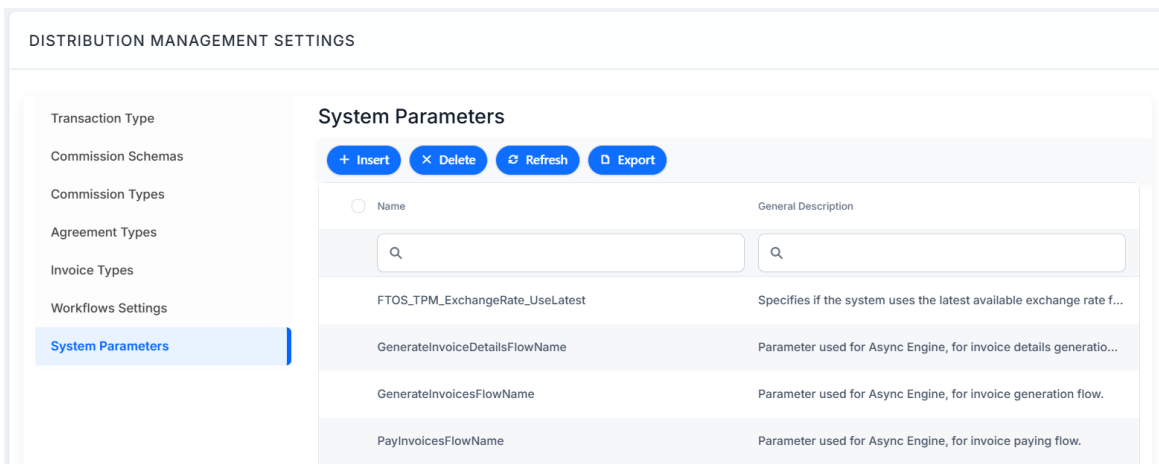
Managing Distribution Management System Parameters

IMPORTANT!

You must have the `system administrator` user right to view and manage the Distribution Management system parameters.

In order to manage the system parameters used by your FintechOS Distribution Management installation, follow these steps:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Settings** > **System Parameters**.
2. On the **Distribution Management System Parameters List** page, you can add new system parameters or search, edit, and delete existing ones.



IMPORTANT!

You can't delete parameters marked as **Is System** or edit anything else except their value.

Creating Distribution Management System Parameters

Follow these steps to create new system parameters to be used with Distribution Management:

1. Click **Insert** on the **System Parameters** page to display the **System Parameter** page.
2. Fill in the following fields:

- **Name** - Enter a suggestive name for the parameter.
 - **Is System** - Select the checkbox to specify that the parameter cannot be deleted or edited, except its value. Leave the checkbox empty if this parameter can be changed or deleted.
 - **General Description** - Enter a detailed description for the system parameter.
 - **Parameter Type** - Select the data type of the parameter. Possible values: Text, Date, Date Time, Invariant Date, Whole Number, Numeric, Option Set and Entity.
3. Fill in the rest of the fields, depending on the selected data type:
 - **Entity** - For Lookup parameter type, select the entity from where you need to pick a record as parameter value.
 - **Option Set** - For Option Set parameter type, select the option set from where you need to pick a value as parameter value.
 - **Parameter Value** - Enter the value of the default parameter. Depending on the selected parameter type, you can either enter a value, select the checkbox or select record:

- For Text, Date, Date Time, Invariant Date, Whole Number, Numeric parameter types, enter the desired value taking in consideration the data type's format.
 - For Boolean parameter types, select the checkbox to specify a True value, or deselect for a False value.
 - For Lookup parameter type, select the desired record from the previously selected entity that acts as parameter value.
 - For Option Set parameter types, select the desired value from the previously selected option set that acts as parameter value.
4. Click the **Save and Reload** button.

External Request Processing

To improve the platform's overall performance through parallel processing, Distribution Management sends the requests generated for processing outside of the platform, instead of being processed within the platform, through JobServer. Thus, Distribution Management processes the requests in a parallel asynchronous way, through [Async Engine](#).

Configuring the Async Engine

The configuration for Distribution Management-Async Engine is as follows:

1. The **GenerateInvoiceDetailsFlowName**, **GenerateInvoicesFlowName** and **PayInvoicesFlowName** Distribution Management system parameters must be used when a request for the specific flow is triggered.
2. The **Async Engine web API client library** has to be loaded on the platform using the Async Engine swagger file.

3. The [Vault settings](#) for the Studio, Portal, and JobServer components must contain a key named **AsyncEngineURL_DistributionManagement** with the value of the Async Engine's URL.

Read the following section for more detailed information:

Calling the External Request Processor

The asynchronous processing configuration is kept in the Vault.

IMPORTANT!

Configure the external request processor app service to use the `async-engine` folder in Vault.

During implementation, create a web API client library to enable Distribution Management to call these configurations, making sure the library uses the latest swagger file found at `<server_address>/services/specifications/swagger-ui`.

Web Api Client Library

Name

Description

Api Type

Min Platform Version

Is Used For Service Pipes

Api Definition

```
openapi: 3.0.0
info:
  version: "1.0"
  title: Async Engine API

paths:
  /async/api/start:
    post:
      tags:
        - Start async engine
      operationId: startProcess
      parameters:
```

IMPORTANT!
Set the Is Used for service pipes attribute to True.

Send the AsyncEngineURL parameter, stored in Vault, with the URL of the external request processor when importing the web api client library in the Distribution Management context.

When the external request processor calls the Portal, the portal user client must be part of the Role defined in rbac.config vault parameter.

IMPORTANT!
Make sure you create a user for running the OpenAPI pipelines. The user must be part of the Registered users group.

While calling the external request processor using the `startProcess` operation, send the following parameters:

- `flowName` - mandatory;
- `correlationId` - optional, automatically generated if missing;
- `payload` - mandatory.

Configuring the Vault Parameters

You must configure a series of parameters so that your external request processor can receive and forward requests.

Add the `AsyncEngineURL_DistributionManagement` Vault parameter to Distribution Management Portal/ Studio/ JobServer. This is used when [importing the web API client library](#).

```
AsyncEngineURL_DistributionManagement
"AsyncEngineURL_DistributionManagement": "https://app-
<async-engine-instance_Distribution>-<project_name>-
<env>.azurewebsites.net"
```

The following parameters are used in the external request processor's Vault parameters:

```
threadpool.config
{
    "poolSize": 10,
    "maxPoolSize": "1000",
    "maxQueueSize": "100"
}
```

```
azure.storage.config
{
    "accountName": "<storage account name>",
    "accountKey": "<storage account key>",
    "suffix": "core.windows.net"
}
```

```

openid.config
{
    "use-resource-role-mappings": "false",
    "autodetect-bearer-only": "true",
    "ssl-required": "external",
    "auth-server-url": "<auth server url>",
    "principal-attribute": "preferred_
username",
    "resource": "<client used for
authentication>",
    "credentials": {
        "secret": "<client secret>"
    },
    "realm": "<realm>"
}

```

```

openapi.url
https://<open api url>.azurewebsites.net/ftosapi/automation-
processors/actions/

```

```

openapi.username
<username>

```

```

openapi.password
<password>

```

```

flow.config
[
    {
        "flowName": "Flow_TPM_GenerateInvoiceDetails_
AsyncEngine",
        "queueMappings": [
            {
                "sourceQueueName": "ftos-tpm-find-
agreements",
                "endpointName": "FTOS_TPM_
FindMatchingAgreements_AsyncEngine",
                "targetQueueName": "ftos-tpm-build-invoice-
details",
                "initial": true
            },
            {
                "sourceQueueName": "ftos-tpm-build-invoice-
details",

```

```

        "endpointName": "FTOS_TPM_
BuildInvoiceDetails_AsyncEngine",
        "targetQueueName": "ftos-tpm-calculate-
pricings"
    },
    {
        "sourceQueueName": "ftos-tpm-calculate-
pricings",
        "endpointName": "FTOS_TPM_CalculatePricings_
AsyncEngine",
        "targetQueueName": "ftos-tpm-generate-
invoice-details"
    },
    {
        "sourceQueueName": "ftos-tpm-generate-
invoice-details",
        "endpointName": "FTOS_TPM_
GenerateInvoiceDetails_AsyncEngine"
    }
]
},
{
    "flowName": "Flow_TPM_GenerateInvoices_AsyncEngine",
    "queueMappings": [
        {
            "sourceQueueName": "ftos-tpm-generate-
invoice",
            "endpointName": "FTOS_TPM_GenerateInvoice_
AsyncEngine",
            "initial": true
        }
    ]
},
{
    "flowName": "Flow_TPM_PayInvoices_AsyncEngine",
    "queueMappings": [
        {
            "sourceQueueName": "ftos-tpm-pay-invoice",
            "endpointName": "FTOS_TPM_PayInvoice_
AsyncEngine",
            "initial": true,
            "maxThreads": 10
        }
    ]
}
]

```

```

queue.config
[
    {
        "name": "ftos-tpm-find-agreements",
        "maxMessages": 10,
        "timeToLive": 1
    },
    {
        "name": "ftos-tpm-build-invoice-details",
        "maxMessages": 10,
        "timeToLive": 1
    },
    {
        "name": "ftos-tpm-calculate-pricings",
        "maxMessages": 10,
        "timeToLive": 1
    },
    {
        "name": "ftos-tpm-generate-invoice-details",
        "maxMessages": 10,
        "timeToLive": 1
    },
    {
        "name": "ftos-tpm-generate-invoice",
        "maxMessages": 10,
        "timeToLive": 1
    },
    {
        "name": "ftos-tpm-pay-invoice",
        "maxMessages": 10,
        "timeToLive": 1
    }
]

```

```

rbac.config
{
    "apiMappings": [
        {
            "url": "/api",
            "roles": [
                "<user profile created and used for service
account roles> "
            ]
        }
    ]
}

```

```

    ],
    "applicationMappings": [
    {
    "url": "/actuator",
    "roles": [
    "<user profile created and used for service
account roles>"
    ]
    },
    ],
    "swaggerMappings": [
    {
    "url": "/specifications",
    "roles": [
    "<user profile created and used for service
account roles>"
    ]
    }
    ]
  }

```

```

redelivery.config
{
  "default": {
    "count": 2,
    "delay": 2
  },
  "additional": [
    {
      "httpStatus": 502,
      "count": 2,
      "delay": 1
    }
  ]
}

```

```

table.config
{
  "statusTable": "statusTableAsyncEngine",
  "dlqTable": "dlqTableAsyncEngine"
}

```

```

http.connection.config
{

```

```
"requestTimeout": "18000",  
"connectionTimeout": "12000",  
"socketTimeout": "23000",  
"totalConnections": "1000000",  
"routeConnections": "500000",  
"connectionTimeToLive": "1000"  
}
```

Managing Unprocessed Requests

If a request is not processed by the Async Engine regardless of the reason, you can [view a message](#) in the dedicated **DLQ dashboard** in your FintechOS Portal. For accessing the dashboard, perform the needed [configurations](#) during the implementation. [Send the request for reprocessing](#) by the Async Engine from the same dashboard, simply clicking a button next to that message.

Configuring the Messages and Access Rights for the DLQ Dashboard

The **DLQ Messages** (dead letter queue) dashboard stores messages about the requests that the Async Engine could not process for any reason. To view the messages and to send the requests back to processing, make sure you perform the following configurations:

1. In FintechOS Studio, select the **Admin > System Parameters** menu.
2. On the newly displayed **System Parameters list** page, find and double-click the **DLQInterfaceQueuesToBeShown** [system parameter](#) to open it for editing. This parameter represents the list of queues from which the messages are displayed in the **DLQ Messages** dashboard in FintechOS Portal.
3. On the **Edit System Parameter** page, edit the parameter value to display the messages from the desired flows. Enter the names of the flows one by one, separated by commas and with no spaces between them. For example:

ftos-cb-loan-approval,ftos-cb-repayment-notification,ftos-cb-bankaccount-transaction,ftos-tpm-find-agreements,ftos-tpm-build-invoice-details,ftos-tpm-calculate-pricings,ftos-tpm-generate-invoice-details,ftos-tpm-generate-invoice,ftos-tpm-pay-invoice,ftos-cb-customer,ftos-cb-customerlimit-create,ftos-cb-customerlimit-approve,ftos-cb-settlementaccount-create,ftos-cb-settlementaccount-approve,ftos-cb-topup-create,ftos-cb-topup-approve,ftos-cb-loancontract-create,ftos-cb-loancontract-approve

The flows must be specified in Vault.

NOTE

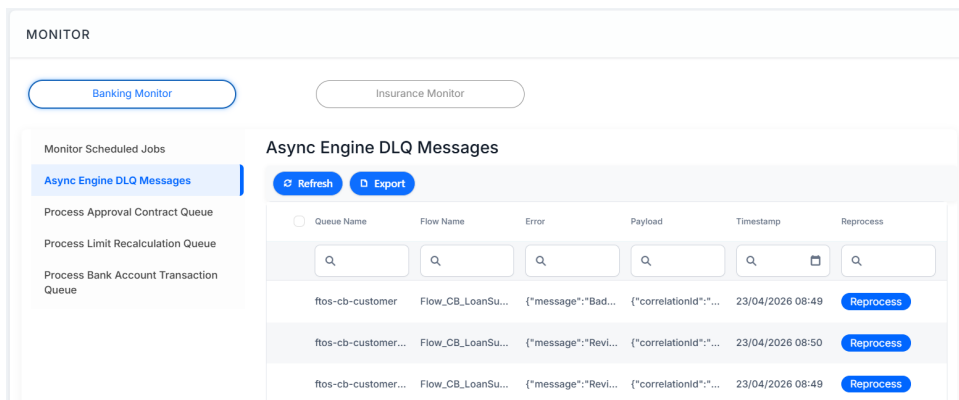
The order of messages displayed in the dashboard depends on the order of the flows you mention here, so make sure you enter the most desired flow name first and then add them according to your priorities.

4. Click the **Save and Close** button.
5. Select the **Security > System Users** menu.
6. On the newly displayed **System Users list** page, find and double-click each user who should have access to the **DLQ Messages** dashboard.
7. [Edit the system users](#), assigning them the **Async Engine Admin** security role.
8. Click the **Save and Close** button.

Viewing the DLQ Messages

Follow these steps to view the requests that the Async Engine could not process:

1. In FintechOS Portal, go to **main menu > Contract Management > Monitor > Banking Monitor > Async Engine DLQ Messages**.



NOTE
Your system user must have the **Async Engine Admin** security role.

2. View the messages in the list, if they exist. Each message contains the queue name, the flow name, the error message, the payload and the time stamp of the unprocessed request.

Reprocessing Failed Requests

You can send for processing each of the failed requests from the **DLQ Messages** dashboard one by one, as follows:

1. Click the **Reprocess** button in line with the desired request.
2. The request is sent back to the Async Engine for processing and an informative message is displayed on your screen.

Scheduled Jobs

Jobs are automated procedures that perform certain tasks, running at a specific time or on a recurring schedule. Read detailed information about scheduling jobs in the FintechOS Studio User Guide's [dedicated page](#).

Distribution Management comes with the following scheduled jobs that perform repetitive procedures:

Generate Invoices Job

The **Generate Invoices** scheduled job runs once each night and finds the relevant details in `Invoice Details` that are the object of an agreement with payment periodicity corresponding to that day, groups them on customer / currency/ invoicing type, and creates an invoice to which attaches all those invoice details. The commission is calculated when each record in `Invoice Detail` is inserted. When using Distribution Management for Banking, each invoice also receives the customer's bank account and reconciliation account, so that the payment can be later performed.

The following service run as part of this job:

1. **Collect Details To Invoice**
The service finds the invoice details that are the object of an agreement with payment periodicity corresponding to that day, groups them on customer / currency/ invoicing type, and creates an invoice to which attaches all those invoice details. When using Distribution Management for Banking, each invoice also receives the customer's bank account and reconciliation account, so that the payment can be later performed.

Charge Not Paid Invoices

The **Charge Not Paid Invoices** scheduled job runs once each night, after the **Generate Invoices** scheduled job, finds all the `Not Paid` invoices with amounts > 0 and changes their status to `Paid`. The update trigger for the

Paid status invoices calls the function defined in the [Invoicing Configurations](#) with **Scope = Payment**. When using Distribution Management for Banking, the defined function performs the transactions needed for the due payments.

The following service run as part of this job:

1. **Charge Not Paid Invoices**

The service finds all the **Not Paid** invoices with amounts > 0 and changes their status to **Pa**id. The update trigger for the **Pa**id status invoices call the function defined in the [Invoicing Configurations](#) with **Scope = Payment**. When using Distribution Management for Banking, it also performs the transactions needed for the due payments.

Security Roles

A security role is a set of privileges and levels of access to various actions/ functions within the FintechOS Platform. Read the [Security Roles topic](#) for detailed information.

Distribution Management has a set of predefined security roles specific for third-party entities management related business needs. Institutions can use these security roles to grant their employees access rights within its systems. Read the [Users topic](#) for information about associating security roles to users in FintechOS Studio.

You can also create new security roles to fit your institution's business needs.

IMPORTANT!

To access Distribution Management features, make sure that your [system users are activated](#), then grant them the [Registered Users default security role](#) and one of the [predefined Distribution Management security roles](#).

Predefined Distribution Management Security Roles

Depending on your installation of Distribution Management, specifically if it is used in conjunction with Loan Management or not, the installation package comes with the following predefined security roles that grant specific access rights to the users associated with one or more of these roles to the FintechOS Platform's actions and functions:

- **Distribution Management Admin** - Users with this security role have read, write, update and delete access rights to the commission schema, commission type and commission records, as well as read, write, update and authorize access rights to third-party agreements and their underlying invoice details.
- **Distribution Management Officer** - Users with this security role have to have read, write and update access rights to third-party agreements and their underlying invoice details.
- **Distribution Management Admin Banking** - This role comes with the Distribution Management for Banking package. Users with this security role have read, write, update and delete access rights to the commission schema, commission type and commission records, as well as read, write, update and authorize access rights to third-party agreements and their underlying invoice details, all in the context of Loan Management business specificity.
- **Distribution Management Officer Banking** - This role comes with the Distribution Management for Banking package. Users with this security role have to have read, write and update access rights to third-party agreements and their underlying invoice details, all in the context of Loan Management business specificity.
- **Async Engine Admin** - Users with this security role can access **Async Engine DLQ Messages** and resend unprocessed requests to the **Async Engine**.

IMPORTANT!

If your user has more than one role, then your access rights are cumulative and you are granted the highest access right.

Read the following sections for detailed information about each security role's access rights to entities and endpoints within the FintechOS Platform:

Distribution Management Admin

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
entity	No	Yes	No	No
PeriodicityType	No	Yes	No	No
Currency	No	Yes	No	No
EntityStatusSettings	No	Yes	No	No
Agreement	Yes	Yes	Yes	Yes
AgreementPricing	Yes	Yes	Yes	Yes
AgreementType	Yes	Yes	Yes	Yes
Commission	Yes	Yes	Yes	Yes
CommissionSchema	Yes	Yes	Yes	Yes
CommissionType	Yes	Yes	Yes	Yes
CommissionValue	Yes	Yes	Yes	Yes
Configuration	Yes	Yes	Yes	Yes
Invoice	Yes	Yes	Yes	Yes
InvoiceDetail	Yes	Yes	Yes	Yes
InvoiceType	Yes	Yes	Yes	Yes
SystemParameter	Yes	Yes	Yes	Yes
optionset	No	Yes	No	No
workflow	No	Yes	No	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_COM_GetPeriodicityTypeInfo
FTOS_TPM_AddInvoiceDetailBtnAction
FTOS_TPM_CalculateInvoiceAmounts
FTOS_TPM_CheckInvoiceHasDetails
FTOS_TPM_CommissionSchemaDetail
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetCommissionInfo
FTOS_TPM_GetDataSourceChartAgreement
FTOS_TPM_GetDefaultAgreementType
FTOS_TPM_GetProductEndpoint

Endpoint
FTOS_TPM_GetProductTypeEndpoint
FTOS_TPM_GetSystemInvariantDate
FTOS_TPM_GetSystemParameter
FTOS_TPM_GetTransactionTypesForClawback

Distribution Management Officer

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	Yes	Yes	Yes	Yes
Currency	Yes	Yes	Yes	Yes
EntityStatusSettings	No	Yes	No	No
Agreement	Yes	Yes	Yes	No
AgreementPricing	Yes	Yes	Yes	No
AgreementType	No	Yes	No	No
Commission	No	Yes	No	No
Invoice	Yes	Yes	Yes	No
InvoiceDetail	Yes	Yes	Yes	No
InvoiceType	No	Yes	No	No
workflow	No	Yes	No	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_TPM_AddInvoiceDetailBtnAction
FTOS_TPM_CalculateInvoiceAmounts
FTOS_TPM_CheckInvoiceHasDetails
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetDataSourceChartAgreement
FTOS_TPM_GetDefaultAgreementType
FTOS_TPM_GetProductEndpoint
FTOS_TPM_GetProductTypeEndpoint
FTOS_TPM_GetSystemInvariantDate
FTOS_TPM_GetSystemParameter
FTOS_TPM_GetTransactionTypesForClawback

Async Engine Admin

A user with this security role can access the **DLQ Messages** dashboard and the following endpoints:

Endpoint
FTOS_AE_GetMessagesFromQueues
FTOS_AE_ReprocessMessage

Distribution Management for Banking security roles

The Distribution Management for Banking package comes with banking-specific security roles which allow users to access more entities needed for Loan Management-specific processes:

Distribution Management Admin Banking

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	Yes	Yes	Yes	Yes
attribute	No	Yes	No	No
entity	No	Yes	No	No
AccountFunctionality	No	Yes	No	No
AccountFunctionality_X_Account	Yes	Yes	Yes	Yes
AccountExtension	Yes	Yes	Yes	Yes
BankAccount	No	Yes	No	No
Contract	No	Yes	No	No
ContractEvent	No	Yes	No	No
Payment	Yes	Yes	Yes	No
PeriodicityType	No	Yes	No	No
AccountType	No	Yes	No	No
Currency	No	Yes	No	No
EntityStatusSettings	No	Yes	No	No
Agreement	Yes	Yes	Yes	Yes
Agreement_BW	Yes	Yes	Yes	Yes
AgreementPricing	Yes	Yes	Yes	Yes

Entity	Insert	Read	Update	Delete
AgreementType	Yes	Yes	Yes	Yes
Commission	Yes	Yes	Yes	Yes
CommissionSchema	Yes	Yes	Yes	Yes
CommissionType	Yes	Yes	Yes	Yes
CommissionValue	Yes	Yes	Yes	Yes
Configuration	Yes	Yes	Yes	Yes
Invoice	Yes	Yes	Yes	Yes
InvoiceDetail	Yes	Yes	Yes	Yes
InvoiceType	Yes	Yes	Yes	Yes
SystemParameter	Yes	Yes	Yes	Yes
optionset	No	Yes	No	No
workflow	No	Yes	No	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter
FTOS_COM_GetPeriodicityTypeInfo
FTOS_COM_GetStatusSettings
FTOS_GetDataSourceChartContract
FTOS_TPM_AddInvoiceDetailBtnAction
FTOS_TPM_CalculateInvoiceAmounts
FTOS_TPM_CheckInvoiceHasDetails
FTOS_TPM_CommissionSchemaDetail
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetBankingProductForPricing
FTOS_TPM_GetBankingProductTypeForPricing
FTOS_TPM_GetCommissionInfo
FTOS_TPM_GetDefaultAgreementType
FTOS_TPM_GetProductEndpoint
FTOS_TPM_GetProductTypeEndpoint
FTOS_TPM_GetSystemInvariantDate
FTOS_TPM_GetSystemParameter
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

Distribution Management Officer Banking

A user with this security role has the following access rights to records in the FintechOS Platform's entities within **their organization**:

Entity	Insert	Read	Update	Delete
Account	Yes	Yes	Yes	Yes
attribute	No	Yes	No	No
AccountFunctionality	No	Yes	No	No
AccountFunctionality_X_Account	Yes	Yes	Yes	Yes
AccountExtension	Yes	Yes	Yes	Yes
BankAccount	No	Yes	No	No
Contract	No	Yes	No	No
ContractEvent	No	Yes	No	No
Payment	Yes	Yes	Yes	No
AccountType	No	Yes	No	No
Currency	No	Yes	No	No
EntityStatusSettings	No	Yes	No	No
Agreement	Yes	Yes	Yes	No
Agreement_BW	Yes	Yes	Yes	Yes
AgreementPricing	Yes	Yes	Yes	No
AgreementType	No	Yes	No	No
Commission	No	Yes	No	No
Configuration	No	Yes	No	No
Invoice	Yes	Yes	Yes	No
InvoiceDetail	Yes	Yes	Yes	No
InvoiceType	No	Yes	No	No
SystemParameter	No	Yes	No	No
optionset	No	Yes	No	No
workflow	No	Yes	No	No

A user with this security role can access the following endpoints:

Endpoint
FTOS_CB_GetDataSourceChartAgreement
FTOS_CB_GetSystemInvariantDate
FTOS_CB_GetSystemParameter

Endpoint
FTOS_COM_GetStatusSettings
FTOS_GetDataSourceChartContract
FTOS_TPM_AddInvoiceDetailBtnAction
FTOS_TPM_CalculateInvoiceAmounts
FTOS_TPM_CheckInvoiceHasDetails
FTOS_TPM_GetAgreementBusinessStatusDisplayName
FTOS_TPM_GetBankingProductForPricing
FTOS_TPM_GetBankingProductTypeForPricing
FTOS_TPM_GetProductEndpoint
FTOS_TPM_GetProductTypeEndpoint
FTOS_TPM_GetSystemInvariantDate
FTOS_TPM_GetSystemParameter
FTOS_TPM_GetTransactionTypesForClawback
FTOS_VerifyUsersCompetence

Commission

Third-party commissions are the fees paid by the your company to third-party entities (agents, brokers, etc.) for intermediating the selling of a product or service to a customer. There may also be commissions paid by the third-party entity to your company, for example for accessing the company's crediting platforms, or even agreement management commissions paid periodically to the company. These third-party commissions vary for each company, based on their policy.

To prevent losing profits, there may be situations when the company claims back all or some of the commission already paid out to third-party entities, because the affected contracts were closed before their due date.

Distribution Management has a dedicated menu for managing third-party commissions. These third-party commissions are attached to agreements with third-party entities.

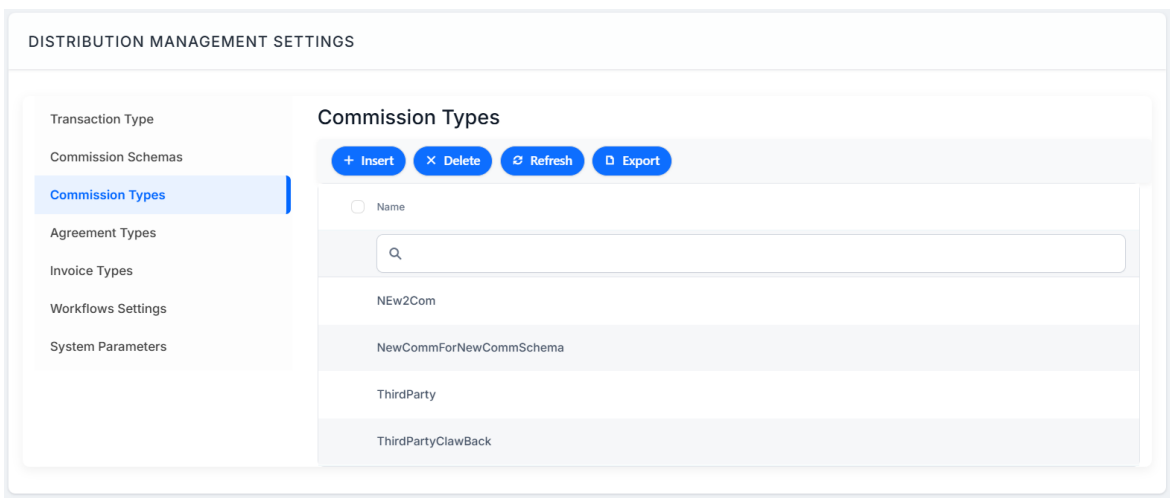
IMPORTANT!

Third-party commissions cannot be used in [contracts](#) for customer, even if using the

Distribution Management for Banking add-on.
They are not displayed in Product Factory's **Commissions** menu and they can't be selected in Commission Lists.
Third-party commissions and third-party commission types can't be used in Payment Schedule Type Details.

To manage third-party commissions:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Settings** > **Commission Types**.
2. On the **Commissions Types** page, you can create a new third-party commission, search, edit, or delete an existing one. You can't delete commissions already used by other records.



IMPORTANT!
Even if using the Distribution Management for Banking add-on, the value of a third-party commission used in active contracts can't be edited. Instead, you can modify the value's validity and add a new value with a future validity period.

NOTE
For information about the available endpoints, please visit the [Distribution Management Endpoints](#) page.

Creating Third-Party Commissions

IMPORTANT! Make sure that the [commission schema](#) and [commission types](#) are already set up before creating your commissions.

Follow these steps to create new third-party commission records:

1. Add Commission Details

1. In the FintechOS Portal, click the **Insert** button on the **Commissions Types** tab. The **Commission Types** page is displayed.

2. Fill in the following fields from the **Main Info** section:
 - **Name** - Enter the name of the commission.
 - **Commission Type** - Choose one from the following commission types:
 - ThirdParty (out-of-the-box third-party commission type)
 - ThirdParty Clawback (out-of-the-box third-party commission type, defined for clawback commissions)
 - Other third-party commission types defined by your users.
 - **Currency** - Select the currency of the commission from the drop-down.

- **Accept Clawback** - Select this checkbox if the commission accepts a clawback commission during agreement pricing definition.
- **Is For Transaction** - Select this checkbox if the commission is applicable when approving a transaction at the contract/ deal level, for example when approving a disbursement. If left unselected, then the commission is applicable when approving the contract/ deal. Depending on this setting, the system automatically triggers the calculation of the commission value in the moment when a contract or a transaction that falls under a third-party agreement is approved.

NOTE

If a clawback commission has **Is For Transaction = True**, the clawback commission is calculated for each transaction type set in the **Clawback Transaction Types** field at the agreement pricing level.

If a clawback commission has **Is For Transaction = False**, the clawback commission is calculated only for the first occurring transaction type from the ones set in the **Clawback Transaction Types** field at the agreement pricing level.

- **Value Is Percentage** - Select this checkbox if the commission is measured by percentage, not as a fixed value
- **Percent Applied To** - Only displayed if you select the checkbox next to the **Value Is Percent** field. Choose one of the following:
 - **Remaining value** - the percentage applies to the contract's remaining to be repaid value .

- **Financed value** - the percentage applies to the contract's financed value.
- **Amount** - the percentage applies to the contract's amount.
- **Interest** - the percentage applies to the contract's interest.

Percent Applied To calculation method for Distribution Management for Banking

For **Term Loan**, **Mortgage** or **Overdraft** banking products the calculation method is as follows:

If `percentAppliedTo = financedAmount`, then
`financedAmount = amountDue - advanceAmount;`

If `percentAppliedTo = amount`, then
`financedAmount = amountDue;`

If `percentAppliedTo = remainingValue`, then, if `Contract Status = ContractVersionDraft`, then
`financedAmount = (-1) * mainBankAccountBalance`. No negative values are allowed, so if the result is negative, then
`financedAmount = null`.

Default value `financedAmount = null`. For **Current Account with Overdraft** banking products the calculation method is as follows:

If `percentAppliedTo = overdraftLimitAmount`, then `financedAmount = overdraftLimitAmount;`

If `percentAppliedTo = usedAmount`, then if `(periodType == Once)`, `financedAmount = overdraftLimitAmount - availableAmountForOverdraft`, else
`financedAmount = null`.

Default value `financedAmount = null`.

3. Click the **Save and Reload** button.

2. Add Commission Values

After saving the commission record, you should define the actual values of the third-party commission. These values are later displayed on the **Commission** page's **Commission Values** section.

To add a new commission value, follow these steps:

1. Click the **Insert** button in the **Commission Values** tab.
2. Fill in the following fields in the newly opened **Commission Value** page:

The screenshot shows a web form titled "Commission Value". On the left, there is a tab labeled "Main Info" with a blue circle icon. To the right of the tab are three input fields, each with a red asterisk above it. The first field is labeled "Commission Percent" and contains the number "5". The second field is labeled "Valid From" and contains the date "01/02/2025". The third field is labeled "Valid To" and contains the date "01/03/2080". Each date field has a small calendar icon to its right.

- **Commission Value** - Enter the value of the commission, if the commission is a fixed value.
- OR
- **Commission Percent** - Enter the percentage representing the commission. If the commission percentage > 100, Distribution Management displays a warning message.
 - **Valid From** and **Valid To** - Select the interval during which the commission value is applicable.
3. Click the **Save and Close** button.

Commission Schema

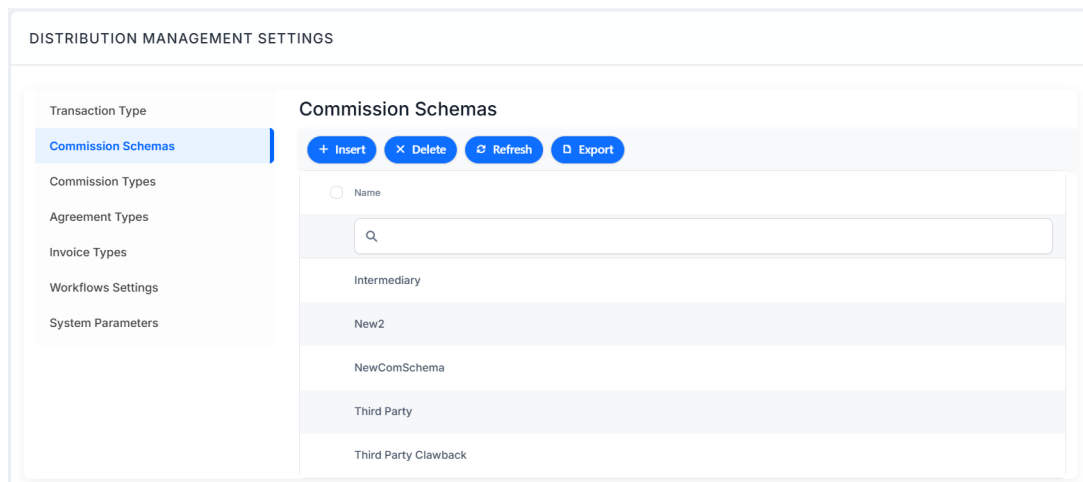
Third-party commission schema records are commission schemas that can be used only for the processes within Distribution Management. They are used to categorize [third-party commission types](#). For example, there are two schemas that come out-of-the-box with Distribution Management: Third Party and Third Party Clawback.

IMPORTANT!

You can add other schemas if needed, just bear in mind that the business logic must also be implemented for your new schemas. By default, the scripts are written to treat only the schemas that come out-of-the-box with the Distribution Management packages, Third Party for commission types given to the third-party entity, and Third Party Clawback for commission types reclaimed from the third-party entity.

To manage third-party commission schemas:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Settings** > **Commission Schemas**.
2. On the **Commission Schemas List** page, you can create a new third-party commission schema, search, edit or delete an existing one. You can't delete schemas already used to define commissions.



Creating Third-Party Commission Schemas

Follow these steps to create new third-party commission schema records:

1. In the FintechOS Portal, click the **Insert** button. The **Commission Schema** page is displayed.

2. Fill in the **Name** field with the name of the third-party commission schema.
3. Click the **Save and Reload** button. The new third-party commission schema is created.
4. Implement the business logic must also be implemented for your new schemas.

Commission Type

Third-party commission types are used to categorize [commissions](#) according to their intended usage. For example, there are third-party type commissions and third-party clawback type commissions.

IMPORTANT!

You can add other third-party commission types if needed, just bear in mind that the business logic must also be implemented for your new third-party commission types.

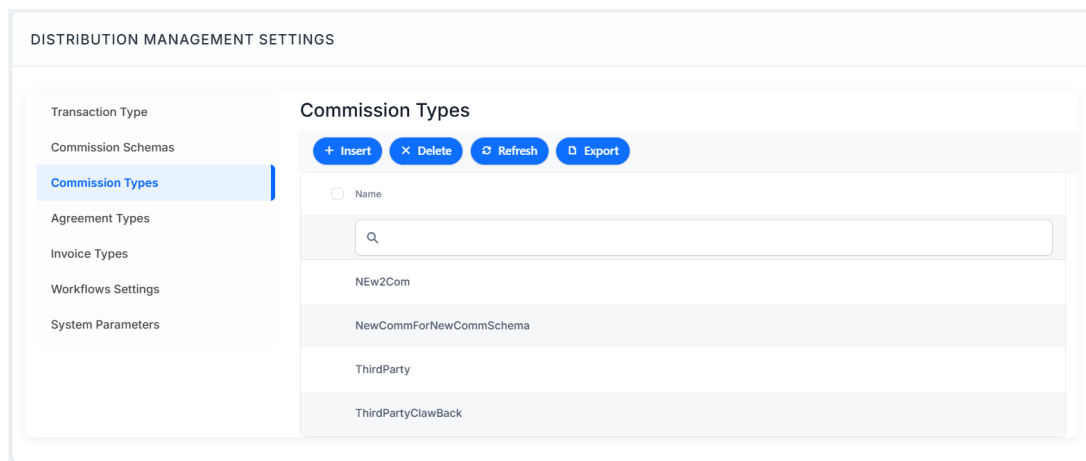
To manage third-party commission types:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Settings** > **Commission Types**.
2. On the **Commission Types List** page, you can create a new third-party commission type, search, edit, or delete an existing one. You can't delete commission types already used to define commissions.

Creating Third-Party Commission Types

Follow these steps to create new third-party commission type records:

1. In the FintechOS Portal, click the **Insert** button. The **Commission Type** page is displayed.



2. Fill in the following fields from the **Main Information** section:
 - **Commission Schema** - Select the schema where this third-party commission type belongs.
 - **Name** - Enter the name of the third-party commission type.
3. Click the **Save and Close** button. The new third-party commission type is created.
4. Implement the business logic must also be implemented for your new commission types.

Managing Third-Party Entities

Third-party entities in a relationship with your institution (merchants, agents, brokers, insurers, etc.) must be registered as customers in your system. A customer is an individual or a legal entity who has business relation with your company. Opening an account is the crucial element in establishing the customer relationship. Create a record for each of the third-party entities you work with and assign them roles. The records are stored in the Account entity.

IMPORTANT!

Perform complex customer management operations using the [Customer 360](#) solutions.

You can also manage customers and their relevant information, especially their roles, from Distribution Management.

Follow these steps to manage third-party entities in Distribution Management:

1. In FintechOS Portal, click the **main menu icon > Lending > Customers**.
2. On the **Customer** page, you can add new third-party entity records or search, edit, assign roles, and delete existing ones.

LENDING CORE PANEL

Customers

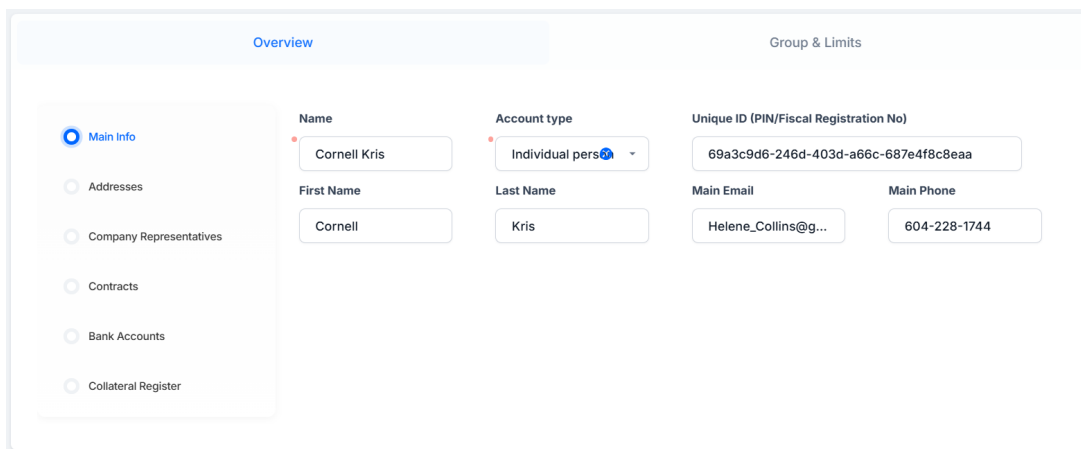
+ Insert X Delete ↻ Refresh 📄 Export

Customer ID	Name	Main Email	Account type	Unique ID (PIN/Fiscal Regi...
11742	Cornell Kris	Helene_Collins@gmail...	Individual person	69a3c9d6-246d-403...
27085	Kellie Moen	Clare.Hauck@hotmail...	Individual person	75ffa934-bbb0-49ad-...
23118	Lilla Fadel	Valentina.Towne@gm...	Individual person	5ea0a71b-59c4-4711-...
21738	Christop Cummerata	Davin34@gmail.com	Individual person	0250

Creating Third-Party Entities & Assigning Roles

Follow these steps to create new third-party entities and assign roles to them:

1. Click **Insert** on the **Customer** page to open the **Overview** page, the first in the third-party entity creation process.



The screenshot displays the 'Overview' page for creating a third-party entity. The page is divided into a sidebar and a main content area. The sidebar on the left contains a list of navigation options: 'Main Info' (selected with a blue circle), 'Addresses', 'Company Representatives', 'Contracts', 'Bank Accounts', and 'Collateral Register'. The main content area is titled 'Group & Limits' and contains several input fields for creating a third-party entity. The fields are arranged in a grid-like structure. The first row contains 'Name' (input field with 'Cornell Kris'), 'Account type' (dropdown menu with 'Individual person' selected), and 'Unique ID (PIN/Fiscal Registration No)' (input field with '69a3c9d6-246d-403d-a66c-687e4f8c8eaa'). The second row contains 'First Name' (input field with 'Cornell'), 'Last Name' (input field with 'Kris'), 'Main Email' (input field with 'Helene_Collins@g...'), and 'Main Phone' (input field with '604-228-1744').

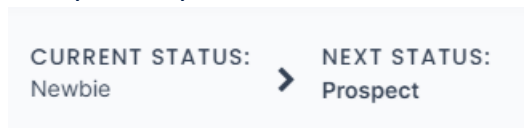
2. Fill in the following fields:
 - **Name** - Enter the name of the third-party entity.
 - **Account Type** - Select the type from the possible options: Legal person, Individual person or Self-employed individual.
3. Optionally, fill in the following fields:
 - **First Name** - Enter the first name of the third-party entity, if this is not a legal person.
 - **Last Name** - Enter the last name of the third-party entity, if this is not a legal person.
 - **Unique ID (PIN/Fiscal Registration No)** - Enter the entity's unique ID:

- For an **Individual** customer enter their personal identification number (PIN).
- For a **Legal** person or a **Self employed individual** entity enter their fiscal registration number.
- **Main Email** - Enter the email address of the third-party entity.
- **Main Phone** - Enter the phone number of the third-party entity.
- **Role** - Select as many roles as you wish for this third-party entity to be able to have within contracts/ deals. In Distribution Management, roles are important when generating invoice details. The values are displayed from the **Contract Role** option set.

NOTE

If used in conjunction with Loan Management, roles can affect the limits that can be set up for a customer. If a customer has a role, then they can have **role-based limits** that are configured with a limit type associated to the same role. For example, if the customer is declared as Merchant, they can have a limit based on a limit type with the associated Merchant role.

4. Click the **Save and Reload** button.
The entity is saved in **Newbie** status, with minimum information, such as an auto-generated customer number. If needed, you can manage detailed customer information using the menus of the **Customer 360** solution.
5. Change the record's status to **Prospect** or **Customer** to be able to select that entity when you create contracts/ deals, as pictured here:



Managing Third-Party Agreements

You can manage third-party agreements via the user interface or via integration through APIs.

NOTE

For information about the available endpoints, please visit the [Distribution Management Endpoints](#) page.

To manage third-party agreements, follow these steps:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Operations** > **Agreements**.
2. On the **Agreements** page, you can [create a new agreement](#), search, edit, or delete agreements in **Draft** status.

DISTRIBUTION MANAGEMENT OPERATIONS

Commissions

Agreements

Agreements Invoices

Customers

Unallocated Invoice Details

Payments

Agreements

+ Insert
× Delete
↻ Refresh
ID Export

<input type="checkbox"/>	Name	Valid From	Valid To	Party	Status	Agreement Type
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
	AG000000094	01/04/2026	01/04/2028	Sawayn Thalia	Approved	Merchant Agree...
	AG000000093	01/04/2026	01/04/2028	Rempel Tina	Approved	Merchant Agree...
	AG000000092	31/03/2026	31/03/2028	Koch Cassie	Approved	Merchant Agree...
	AG000000091	31/03/2026	31/03/2028	Kuhn Sharon	Approved	Merchant Agree...

IMPORTANT!

Users with the associated role of **Distribution Management Admin** or **Distribution Management Admin Banking** can view, insert, update, and authorize third-party agreement records.

Users with the associated role of **Distribution Management Officer** or **Distribution**

Management Officer Banking can view, insert, and update such records, but they can't authorize them.

NOTE

You can also manage agreements in the **Third-Party Agreements dashboard**. When used in conjunction with Loan Management, agreements that remain in Draft status for a predefined number of days can be purged within the **Records To Be Purged dashboard's Agreements** tab.

Third-Party Agreements Life Cycle and States

Third-party agreements are complex agreements between your company and third-party entities such as agents or brokers, who intermediate the selling of products to customers, in exchange of a payment previously negotiated the your company. Therefore the four-eyes principle is applicable here, meaning that a record should be approved by a second company employee, with higher authorization rights. This is enabled via approval task FintechOS Platform capabilities and thus it is also a company's responsibility to set proper security roles and access rights to its users, in order to make sure that the same user can't insert and also authorize the same record.

A third-party agreement record has the following business workflow statuses:

- **Draft** - the status of a newly created third-party agreement record that was not yet sent for approval. While in this status, you can edit the fields from the record's **Agreement** tab, but Distribution Management doesn't attach any invoices to it. Send the record to approval after editing all the necessary details and adding at least one agreement pricing record.
- **Pending** - this is a system status applied to records sent for approval, but not yet approved (when the four-eyes approval process is implemented). You can't perform any updates in this system status.

- **Approved** - the status of a third-party agreement record after being authorized by a user with approval competencies. While in this status, you can't edit the record's details, but the invoice details are automatically added through the Distribution Management invoicing processes. If you need to alter the record's details, create a new version based on the current agreement.
- **Closed** - the last status of a third-party agreement, after manually closing it or after creating a new version based on the current version. You can't perform any updates on the record.
- **Canceled** - the status of a record after manually canceling it straight from the Draft status. You can't perform any updates on the record.

Third-Party Agreements Versioning

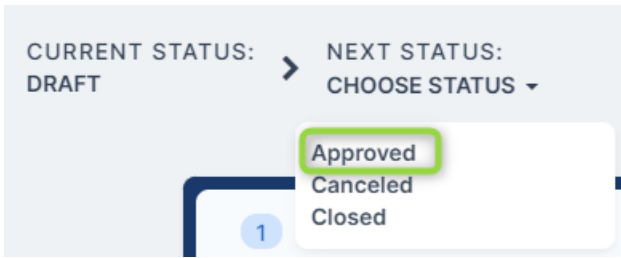
Distribution Management allows you to [create new versions for an existing agreement](#) if you need to modify an existing approved record.

A third-party agreement version can have the following statuses:

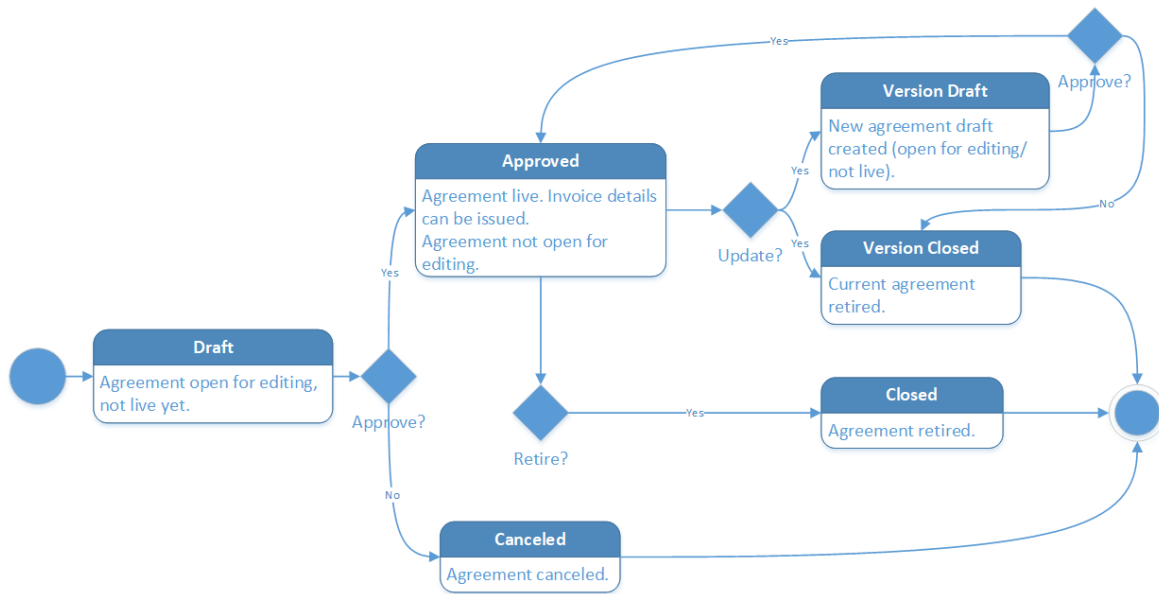
- **Version Draft** - the status of a newly created agreement version record that was not yet sent for approval. While in this status, you can edit some fields. Send the record to approval after editing all the necessary details.
- **Approved** - the status of an agreement version record after being authorized by a user with approval competencies. While in this status, you can't edit the record's details.
- **Version Closed** - the last status of an agreement version, after manually closing it or after creating another new version based on the current version. You can't perform any updates on the record.

Changing Third-Party Agreement Statuses

You can manage a third-party agreement's life-cycle by changing its status from the top right corner of the screen.



The third-party agreement status transitions are illustrated below:



Note that:

- Once a record is live, its settings can no longer be modified.
- If you want to update the details of a live agreement, you must create a new agreement version.
- When you create a new agreement version, the current version is retired; no updates are allowed on the retired version.
- Every agreement version starts in a Draft state and must go through an approval process before going live.
- Only one version of an agreement can be live at one time.

NOTE

As a best practice, new records or new versions of existing records created on a specific day should be approved on the same day.

Creating Agreements For Third-Parties

Third-party agreements are complex agreements between a company and third-party entities such as merchants, agents, or brokers, who intermediate the selling of products to customers, in exchange of a payment previously negotiated with the company.

Before creating an agreement for a third-party entity, make sure that:

- the [third-party roles](#) are set up according to your company's needs,
- the third-party entity is recorded as a customer in Distribution Management,
- the third-party has the desired [role associated within its customer record](#).

NOTE

When using Distribution Management for Banking, also make sure that the [role-based limits](#) are set up according to your financial institution's needs, and that a settlement account (a current account contract for the same third-party entity) exists either externally, or is set up within Loan Management for the desired currency.

To create a new third-party agreement:

1. Add Agreement Details

1. Open the **Agreements** page as described on the [Managing Third-Party Agreements](#) page.

- Click the **Insert** button to open the **Agreement** page, with the first tab displayed, the **Agreement** tab.

The fields displayed here are also available for completion when updating a record in **Draft** status.

- Fill in the following fields in the **Main Info** section:
 - Party** - Select from the list the name of the third-party entity (merchant, agent, broker, insurer, etc.) with whom the agreement is created. This is the entity who should be mentioned in contracts/ deals as participant with the specified role in order to qualify for the commissions stated in the agreement pricing records added to this agreement. The list is already filtered, displaying only the customers that have the same role as the one selected in the **Role** field.

NOTE

Once an agreement is saved, you can only change the **Role** and **Party** fields if you [create a new version for the agreement](#).

- Role** - Select from the drop-down the role of the entity with whom the agreement is created. The roles are listed in the `ThirdPartyRoleDistribution Management` system parameter. The default roles are: Agent, Broker, Insurer, and Merchant.

- **Agreement Type** - Select the agreement type applicable for this agreement. It is automatically filled in with the **agreement type marked as default**.
- **Agreement Period** - Enter the number of months for the agreement's validity.
If you modify the **Valid To** date, then the value of **Agreement Period** is recalculated, rounding up the fractions of a month to 1 whole month.
- **Valid From** - Select the date when the agreement becomes active.
The maturity date is automatically calculated following the formula:
 $\text{Agreement Date} + (\text{Period} * \text{Period Type})$.
- **Valid To** - The date until when the agreement is valid. This field is automatically completed with the date calculated as the **Valid From** date + the number of months entered in the **Agreement Period** field. You can modify this date from the attached calendar, if needed, but it must be greater than or equal with **Valid From**. If you modify this date, then the value of **Agreement Period** is recalculated, rounding up the fractions of a month to 1 whole month.
- **Agreement Date** - Select the date of the agreement.
- **Currency** - Select from the list the currency of the agreement. This is the currency for settling the included commissions. Make sure the third-party has a current account in the selected currency.
- **External Identifier** - Enter an external identifier of the agreement record, if available.
- **Invoicing Type** - Select the invoicing type applicable for this agreement.
- **Payment Periodicity** - Select from the list the periodicity for processing the payments calculated based on this agreement. The possible values are:

- **Daily** - the payments are performed each day.
- **Weekly** - the payments are performed once a week. If you select this option, the **Week Day** field is displayed.
- **Monthly** - the payments are performed once a month. If you select this option, the **Day of Payment** field is displayed.
- **Settlement Account** - (Only visible when using the Distribution Management for Banking). Select the entity's bank account that acts as settlement account, where the commissions payable based on this agreement should be disbursed and/ or from where your company should subtract the amounts to recover. If used in conjunction with Loan Management, the list is already filtered to display only the selected customer's current accounts in **Open** status, in the currency selected previously for the agreement.
- **Conditions** - Enter any other conditions applicable to the agreement.

4. Click the **Save and Reload** button.

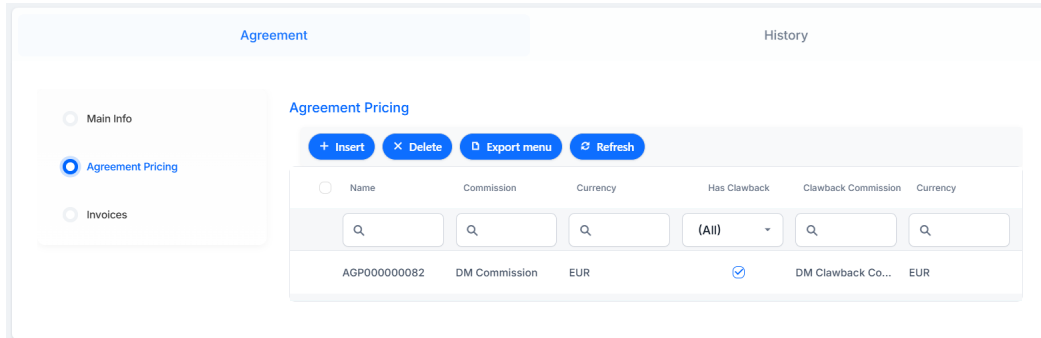
After saving the agreement, a unique identifier is generated based on the increment number and is displayed as the name of the agreement at the top of the page, along with the versioning information. All the other sections of the **Agreement** page (**Agreement Pricing** and **Invoices**) become visible after saving the record, so that you can fill them in.

CURRENT STATUS: DRAFT	>	NEXT STATUS: CHOOSE STATUS -	NAME AG000000407	INCREMENT NO 407	VERSION 1	VERSION DATE 23/03/2023 02:00
--------------------------	---	---------------------------------	---------------------	---------------------	--------------	----------------------------------

The record is still in **Draft** status and you should define at least one [pricing](#) record before you change its status to **Approved**. You or Distribution Management's automatic process can [create invoices](#) only for agreement records with **Approved** status.

2. Define Agreement Pricing

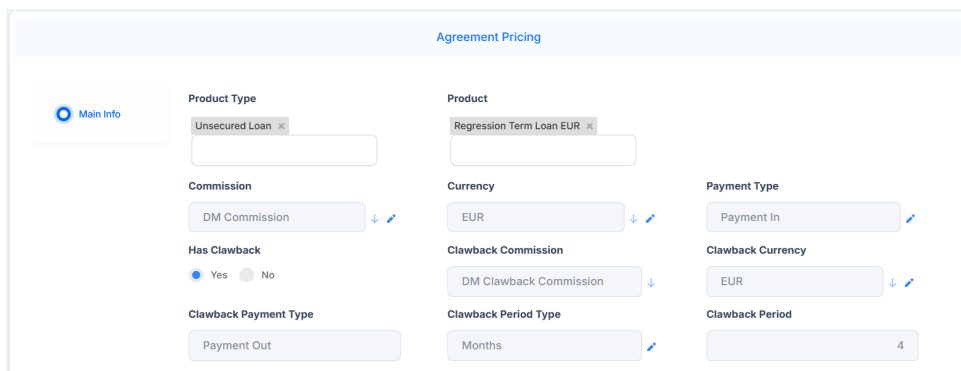
You can insert, update, delete, or export pricing records for an agreement in **Draft** status in the **Agreement Pricing** section.



The section displays information about the pricing's commission and its currency, the clawback commission, if applicable, and the products for which the pricing was set up. You can add as many pricing records as you need for an agreement.

To add a pricing record, follow these steps:

1. Click the **Insert** button to display the **Agreement Pricing** page.



2. Fill in or view the following fields within the newly opened page:
 - **Product Type** or **Product** - Select the product type or the product that must be present in a contract in order for the agreement pricing to be applicable.

NOTE

If you are planning to use the clawback settings, make sure

that you mark the desired transaction types with `Is Clawback Transaction = True`, and those transaction types are selected at the banking product type level.

- **Commission** - Select from the list the third-party commission to be applied for the agreement pricing. The list is filtered to display only commissions with `ThirdParty` commission schema.
 - **Payment Type** - Select the type of payment to be performed for the commission:
 - `Payment In` - for payments from the third-party's settlement account into the financial institution's reconciliation account
 - `Payment Out` - for payments from the financial institution's reconciliation account into the third-party's settlement account.
 - **Currency** - This field is automatically completed with the selected commission's currency. You can't change this value.
3. Click the **Save and Reload** button. After saving the agreement pricing, a unique identifier is generated for it and is displayed at the top of the page, along with the agreement information.
 4. Decide whether or not the pricing has a clawback attached to it. To prevent losing profits, there may be situations when your company claims back all or some of the commission already paid out to third-party entities, due to the fact that the affected contracts/ deals were closed before their due date through certain events that determine the clawback (for example, Early Repayment or Returned Amount or Goods events).
The **Has Clawback** checkbox is displayed only if the selected commission was defined with `Accept Clawback = True`. If you select this checkbox, then the pricing has a clawback commission attached to it. The fields within the **Clawback Settings** section are

displayed only if `Has Clawback = True`. Default value: `False`.

5. In the **Clawback Settings** section, fill in or view the following fields:
 - **Clawback Commission** - If displayed, select from the list the third-party clawback commission to be applied for the agreement pricing. The list is filtered to display only commissions with `ThirdParty Clawback` commission schema.
 - **Clawback Currency** - If displayed, this field is automatically completed with the selected clawback commission's currency. You can't change this value.
 - **Clawback Payment Type** - If displayed, this field is automatically completed with the type of payment to be performed for the clawback commission:
 - `Payment In` - for payments from the third-party's settlement account into your company's reconciliation account;
 - `Payment Out` - for payments from your company's reconciliation account into the third-party's settlement account. You can't change this value.
 - **Clawback Period Type** - If displayed, select from the list the period type for the clawback commission.
 - **Clawback Period** - If displayed, enter the number of periods during which the clawback commission can be reclaimed by the financial institution in case a contract/ deal subject to the agreement pricing was closed earlier or a transaction type marked as clawback transaction type was performed on such a contract/ deal.
 - **Clawback Transaction Types** - Select the transaction types for which the clawback commission should be applied. You can choose from the list of transaction types with `Is Clawback Transaction = True`, based on the responses from the API

Product Type and API Product custom-built endpoints defined during [agreement type](#) creation.

6. Click the **Save and Reload** button.
7. Select the **Block After Clawback** checkbox to mark the contract/ deals as blocked for further agreements after performing a clawback payment. If True, which is the default value, when a clawback commission is paid for this contract/ deal, no other commission found on the contract/ deal can be invoiced by Distribution Management.
8. Click the **Save and Close** button.

After defining the relevant details of the agreement and at least one pricing record, proceed to [agreement approval](#).

IMPORTANT!

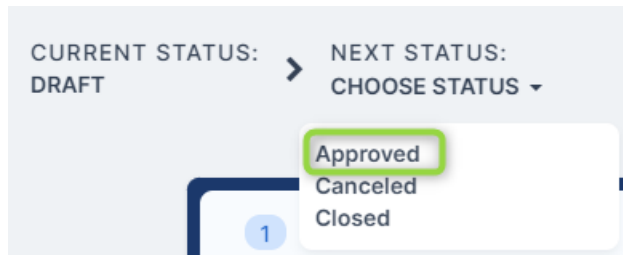
You or Distribution Management's automatic process can [create invoices](#) only for agreement records with **Approved** status. These invoices are later used for paying out the commissions to the third-party entity and/ or your company.

Approving a Third-Party Agreement

You can perform the approval from the Distribution Management user interface.

After defining the relevant details of the agreement and at least one pricing record, proceed to agreement approval:

1. Select an agreement in **Draft** (or **Version Draft**) status.
2. Change its status into **Approved**.



3. Click **Yes** to confirm your action.

Are you sure that you want to change the business status?



If Distribution Management performs all the validations successfully, then the current status of the agreement changes to **Approved**. While in this status, you can't edit the record's details, but the invoice details are automatically added through the [Distribution Management invoicing processes](#). If you need to alter the record's details, [create a new version](#) based on the current agreement.

NOTE

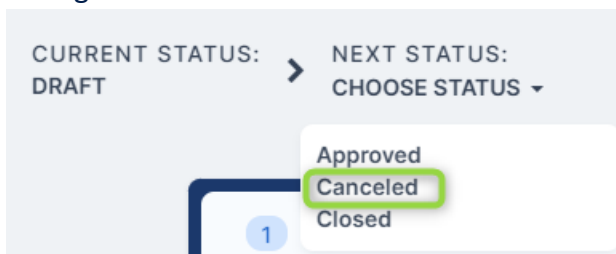
If Distribution Management is used in conjunction with Loan Management, then Loan Management validates the existence of a [setting for a reconciliation account](#) for a specific currency upon third-party agreement or agreement version approval. It also checks whether the reconciliation account setting has continuity for the entire validity period of the agreement.

Rejecting a Third-Party Agreement

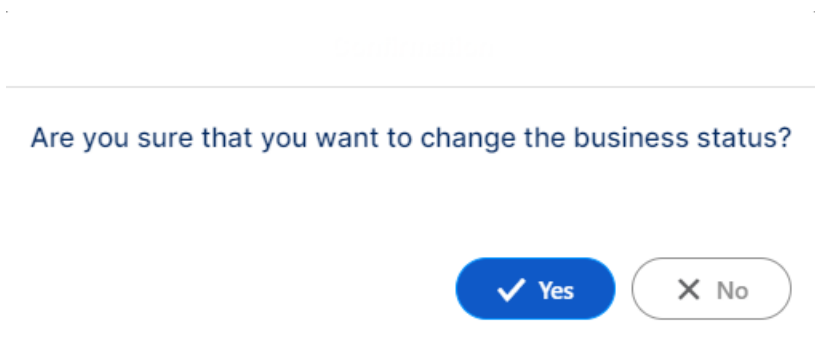
You can reject an agreement, canceling it, when the deal with the third-party entity drops. You can perform the cancellation from the Distribution Management user interface.

Follow these steps to cancel the agreement:

1. Select an agreement in **Draft** (or **Version Draft**) status.
2. Change its status into **Canceled**.



3. Click **Yes** to confirm your action.



If Distribution Management performs all the validations successfully, then the current status of the agreement changes to **Canceled**.

NOTE

You can't further use a canceled agreement. Create a new agreement, if you need to.

Creating New Versions Of Third-Party Agreements

The third-party agreements are [set up for versioning](#). Thus, if you have to update the details of an approved agreement, then you must create a new version of the record.

To create a new version for a record with the **Approved** status, follow these steps:

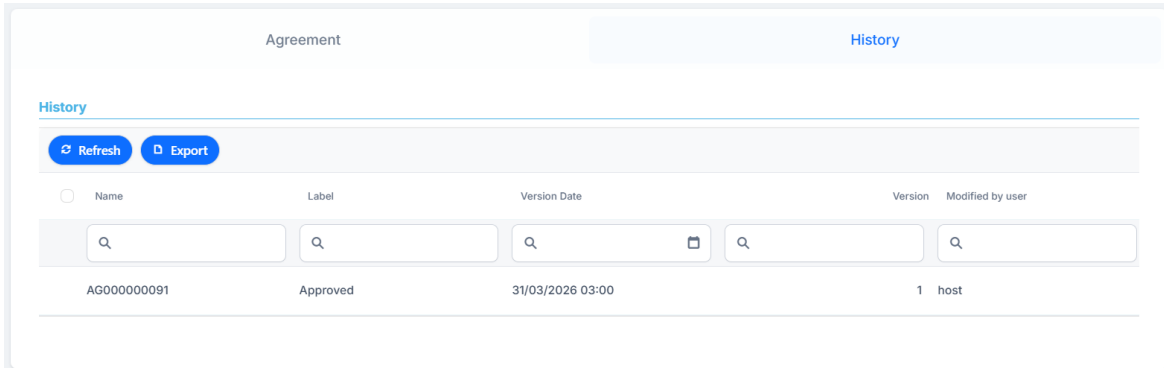
1. Double-click the agreement record selected for updates.
2. Click the **New Version** button in the top right corner of the page.
A new version of the agreement is created, with **Version Draft** status, thus restarting the life-cycle.
3. Edit the desired fields in the **Agreement** tab.
4. Click the **Save and Reload** button.

If the **Version Draft** record is approved, then the original record transitions into the **Version Closed** status and the secondary version becomes the **Approved** currently active agreement record.

Read more details about versioning a record on the [How to Version an Entity Record](#) page.

Viewing a Third-Party Agreement's History

You can view the versions of the agreement, their workflow status, the version creation date, and the user who modified the record, in the third-party agreement's **History** tab. This tab only appears after saving the agreement record.



Here you can track the record's life-cycle and review older versions that are no longer active (for details, see [Third-Party Agreements Statuses](#)).

There are no edits allowed in this tab. To view a version of the record, double-click it.

Working with Third-Party Invoices

Third-party invoices are the invoices that track the incomes and expenses resulted from contracts/ deals based on the institution's [agreements with third-party entities](#) (merchants, agents, brokers, etc.) . Distribution Management has a dedicated menu for managing third-party invoices. These third-party invoices are attached to agreements with third-party entities. You can [create invoices manually](#), or allow Distribution Management to [create the invoices automatically](#).

NOTE

The commissions applicable to agreement pricings are automatically calculated by the system and recorded in the [invoice details](#). The commission calculation is triggered either when a the contract/ deal or an event at the contract/ deal level is approved. The commission is calculated by sending a request to an [external request processor](#) function (for example the Azure function) corresponding to the [ProcessFunc-TPM-FindAgreements](#) system parameter. The function calls a series of endpoints within the system that perform the commission calculation and invoice detail records insertion. These calculated invoice details are later attached to invoices, either automatically or manually.

For Distribution Management for Banking, the commission is automatically calculated when a contract event or the contract itself is approved (for agreement pricings with `Is For Transaction = True`).

A [dedicated job](#), based on the third-party agreement's `Payment Periodicity` (daily, monthly, or weekly), creates invoices and attaches their corresponding invoice details.

NOTE

For information about the available endpoints, please visit the [Distribution Management Endpoints](#) page.

Managing Invoices For Third-Parties

To manage third-party invoices:

1. In FintechOS Portal, click the main menu icon > **Distribution Management** > **Distribution Management Operations** > **Agreements Invoices**.
2. On the **Agreements List** page, you can [create a new invoice manually](#), search, edit, or delete an existing third-party invoice in **Draft** status.

DISTRIBUTION MANAGEMENT OPERATIONS

Commissions

Agreements

Agreements Invoices

Customers

Unallocated Invoice Details

Payments

Agreements Invoices

+ Insert
× Delete
↻ Refresh
📄 Export

<input type="checkbox"/>	Name	Account	Currency	Total Amount T...	Total Amount T...	Start Period	End Period	Business Str
<input type="checkbox"/>	TPMI00000...	Bartoletti Go...	EUR	120.00	200.00	06/03/2026	06/03/2026	invoicePai
<input type="checkbox"/>	TPMI00000...	Nicolas Crist...	EUR	0.00	200.00	22/11/2024	22/11/2024	invoicePai
<input type="checkbox"/>	TPMI00000...	Rau Jermain	EUR	0.00	200.00	22/11/2024	22/11/2024	invoicePai
<input type="checkbox"/>	TPMI00000...	Sawayn Thalia	EUR	0.00	200.00	01/04/2026	01/04/2026	invoicePai
<input type="checkbox"/>	TPMI00000...	Orn Grant	EUR	120.00	200.00	23/01/2025	23/01/2025	invoicePai

NOTE

Distribution Management can also [create the invoices automatically](#), using the [Generate Invoices scheduled job](#). This job runs once each night and creates third-party invoices and payments, for the combination of third-party entity/ agreements currency, during the validity of the agreement, on the Payment Day of each agreement, as defined in the third-party agreement's Payment Periodicity (daily, monthly, or weekly).

IMPORTANT!

Users with the associated role of **Distribution Management Admin** or **Distribution Management Admin Banking** can view, insert, update, and authorize third-party invoice records.

Users with the associated role of **Distribution Management Officer** or **Distribution**

Management Officer Banking can view, insert, and update such records, but they can't authorize them.

Third-Party Invoices Life Cycle and States

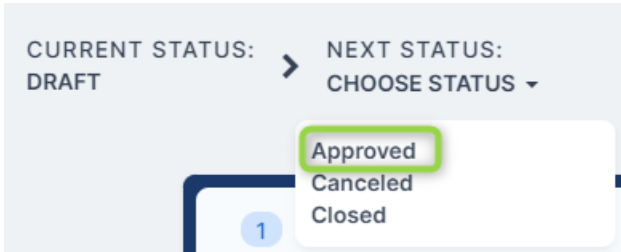
The four-eyes principle is applicable for status transitions of a third-party invoice, meaning that a record should be approved by a second employee, with higher authorization rights. This is enabled via approval task FintechOS Platform capabilities and thus it is also the institution's responsibility to set proper security roles and access rights to its users, in order to make sure that the same user can't insert and also authorize the same record.

A third-party invoice record has the following business workflow statuses:

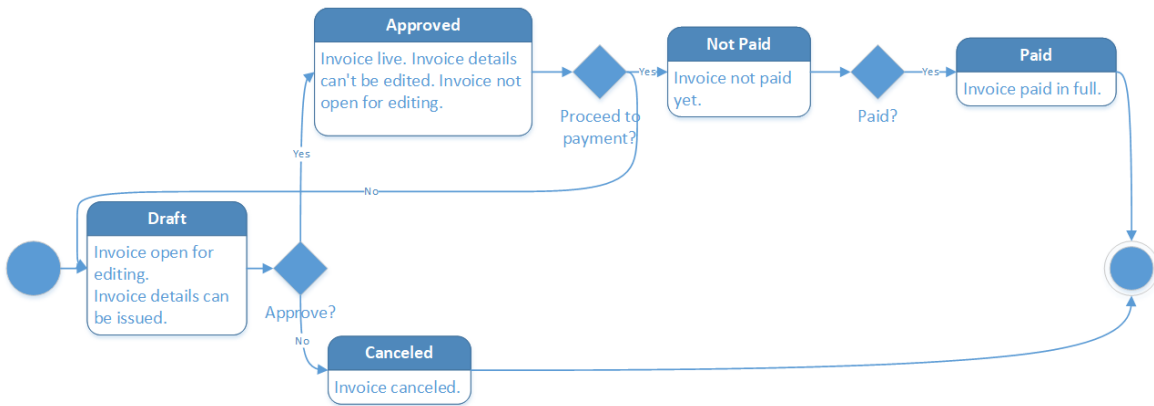
- **Draft** - the status of a newly created invoice record that was not yet sent for approval. While in this status, you can edit the fields from the record's **Invoice** and **Invoice Details** tabs, but no payments can be processed yet. Send the record to approval after editing all the necessary information and adding at least one invoice detail record.
- **Approved** - the status of an invoice record after being authorized by a user with approval competencies. While in this status, you can't edit the record's details. If you need to alter the record's details, change its status back to **Draft**.
- **Not Paid** - the status of an invoice record after approval and before actually performing the bank account transactions for the due payments.
- **Paid** - the last status of an invoice record after performing the bank account transactions for the due payments. No other transitions are allowed from this status.
- **Canceled** - the last status of a record after manually canceling it straight from the **Draft** status. No updates are allowed on the record. No other transitions are allowed from this status.

Changing Third-Party Invoice Statuses

You can manage a third-party invoice's life-cycle by changing its status from the top left corner of the screen.



The third-party invoice status transitions are illustrated below:



Creating Third-Party Invoices

Third-party invoices are the invoices that track the incomes and expenses resulted from contracts/ deals based on the institution's [agreements with third-party entities](#) (agents, brokers, etc.) . Allow Distribution Management to [create the invoices automatically](#), or [create invoices manually](#).

NOTE

The commissions applicable to agreement pricings are automatically calculated by the system and recorded in the [invoice details](#). The commission calculation is triggered either when a the contract/ deal or an event at the contract/ deal level is

approved. The commission is calculated by sending a request to an [external request processor](#) function (for example the Azure function) corresponding to the [ProcessFunc-TPM-FindAgreements](#) system parameter. The function calls a series of endpoints within the system that perform the commission calculation and invoice detail records insertion. These calculated invoice details are later attached to invoices, either automatically or manually.

For Distribution Management for Banking, the commission is automatically calculated when a contract event or the contract itself is approved (for agreement pricings with `Is For Transaction = True`).

A [dedicated job](#), based on the third-party agreement's `Payment Periodicity` (daily, monthly, or weekly), creates invoices and attaches their corresponding invoice details.

Automatically Create Invoices

Distribution Management automatically creates invoices for approved third-party agreements, using a [dedicated scheduled job](#), both for `PaymentOut` and for `PaymentIn`, based on the third-party agreement's `Payment Periodicity` (daily, monthly, or weekly). The automatic invoices contain the same information as the manually added ones, with the difference that `Start Period = End Period = current system date`, and the `Payment Mode = Automatic`. The job runs once each night and creates third-party invoices, for the combination of third-party entity/ agreements currency, during the validity of the agreement, on the `Payment Day` of each agreement, as defined in the third-party agreement's `Payment Periodicity` (daily, monthly, or weekly). The job also attaches invoice detail records.

To prevent losing profits, there may be situations when your institution claims back all or some of the commission already paid out to third-party entities, because the affected contracts were closed before their due date. Distribution Management automatically identifies the transactions that determine the clawback (for example, `Early Repayment` events). To make use of the clawback settings, make sure that you mark the desired [transaction types](#) with `Is Clawback Transaction = True`.

NOTE For Distribution Management for Banking, make sure that those transaction types are selected at the product type level or at the product level, depending on your [agreement pricing definition](#).

Distribution Management calculates the clawback commission and inserts a corresponding invoice detail for the invoice generated for the third-party entity. If no invoice was yet generated for the third-party entity, a new invoice is created, and the invoice details are included there.

NOTE

If a contract/ deal is marked as blocked for further agreements after performing a clawback payment (`Block After Clawback = True`), when a clawback commission is paid for this contract, no other third-party commission found on the contract can be invoiced by the system, hence, for the same agreement pricing, no future invoice details are generated.

For any type of transaction performed on the contracts/ deals subject to agreements that have `Has Clawback = True`, if the Clawback Period covers the date of the transaction and the contracts/ deals are closed, then the system automatically generates an invoice so that the third-party entity is charged.

Example of Clawback Commission applied by Distribution Management for Banking

Let's say you have a transaction that has `Has Clawback = True`, it is included in the product that is captured in the **Agreement Pricing Object** section of the agreement, and it is also captured in the **Clawback Settings'** `Clawback transaction types` field. When a contract is created for one of these products including the specified transaction, Distribution Management commissions the third-party entity as defined within the agreement. If the transaction targeted for clawback happens within the time frame resulting from clawback period type and period, Distribution Management automatically generates an invoice for the clawback commission.

For a setup like the above, if all the conditions are met and a contract for a Buy Now Pay Later product type is included in the invoice for the agreement and if an Early Repayment transaction is performed within the first 3 months from creating the contract/ generating the initial commission, then Distribution Management triggers the clawback automatically.

Manually Create Invoices

Using Distribution Management's user interface, you can create third-party invoices from a dedicated menu item, or directly from an approved agreement's page, with a few differences for each method, as follows:

Creating Invoices Using a Dedicated Menu

Follow these steps to create new third-party invoice records for any approved agreement:

1. In the FintechOS Portal, click the **Insert** button on the top right side of the **Invoices** page (for Distribution Management for Banking, use the **Agreement Invoice** page). The **Invoice** page is displayed.

NOTE The Invoice page for Distribution Management for Banking has a few differences specific for the banking business. They are mentioned below, where needed.

The screenshot shows the 'Invoice' form with the following fields and values:

General Data		Period		
Name	External Identifier	Invoice Date	Start Period	End Period
		2023/04/07	2023/03/26	2023/04/07
Invoicing Type	Payment Mode			
Merchant Inv...	Automatic			
Payment Data		Account Data		
Currency	Total Amount To Pay	Total Amount To Recover	Party	
EUR				

Invoice Details

2. In the **General Data** section of the newly displayed page, fill in, view or modify the following fields:
 - **Name** - Automatically filled in with the name of the invoice record, after saving the record. You can't modify this field.
 - **External Identifier** - Enter an external identifier for the invoice, if needed.
 - **Invoice Date** - Automatically filled in with the current date of the system. You can modify this date.
3. Select the **Invoicing Type** for this invoice from the list of [configured invoice types](#).
4. Select the **Payment Mode** for this invoice. You can select one of the following options:
 - **Automatic** - The entire invoice payment process is managed automatically by [a dedicated job](#).
 - **Triggered** - The invoice payment process is triggered via the FTOS_TPM_PayInvoice endpoint.
 - **Manual** - The invoice payment process must be started manually from the user interface, changing the invoice's status from Not Paid to Paid.

NOTE

Regardless of how it is triggered, rest of the payment process is automatic, as follows: the FTOS_TPM_PayInvoice endpoint changes to invoice's business status to Paid. Then, [the function configured](#) for the Payment scope performs the payment. For Distribution Management for Banking, the function performs the bank account transactions needed to perform the invoice payment.

5. In the **Period** section, fill in the following fields:
 - **Start Period** and **End Period** - Select the starting and the ending date of the interval during which Distribution Management filters the contracts/ deals that are subject to invoicing. The Start Period must be \leq End Period.
6. In the **Payment Data** section, fill in or view the following fields:
 - **Currency** - Select the currency in which the invoice is to be paid.
 - **Total Amount To Pay** and **Total Amount To Recover** - The total amounts to pay to and to recover from the third-party entity, expressed in the selected currency. You can't modify these fields. They are automatically calculated by Distribution Management based on the invoice details attached later, after saving the invoice record.
7. In the **Account Data** section, fill in the following fields:

- **Party** - Select the third-party entity for whom you are creating the invoice.
8. Click the **Save and Reload** button.
 9. For Distribution Management for Banking, also fill in these fields:
 - **Bank Account** - Select the third-party entity's settlement account. You can only choose from the list of the entity's accounts opened in the selected currency.
 - **Reconciliation Account** - Select your institution's reconciliation account. You can only choose from the list of the accounts opened in the selected currency.
 10. Click the **Save and Reload** button.

After saving the invoice record, the **Invoice** page also displays the **Invoice Details** section, containing a list with all the invoice details attached to the invoice, and the **Payments** section, containing information about the [payments performed by Distribution Management](#).
 11. Add invoice details to the invoice in **Draft** status. Click the **Generate Invoice Details** button to automatically attach the details calculated by Distribution Management.
 12. Or, you can attach the details manually, clicking **Insert** within the **Invoice Details** section on the **Invoice** page.

The **Invoice Detail** page is displayed.

 - On the **Invoice Detail** page's **General Data** section, view the following automatically filled in fields: **Invoice** and **Agreement**.
 - In the **Settings** section, view the following automatically filled in fields: **Detail Type**, **Contract**, and **Contract Event**.
 - In the **Payments** section, view in the following automatically filled in fields: **Commission**, **Currency**, and **Exchange Rate**, the exchange rate for Commission Currency to Invoice Currency

valid on the date of the invoice, or the latest exchange rate recorded for Commission Currency to Invoice Currency.

- In the **Values** section, view in the following automatically filled in fields: **Value**, the calculated commission value, and **Equivalent Value**, commission's equivalent value in the invoice's currency, calculated by Distribution Management as (Exchange Rate * Value).

NOTE

Equivalent Value is summed up to Total Amount To Pay if Detail Type = Payment Out.

Equivalent Value is summed up to Total Amount To Recover if Detail Type = Payment In.

13. Click the **Save and Close** button.
14. Change the status of the invoice record to **Approved** to allow Distribution Management to automatically process the payments.

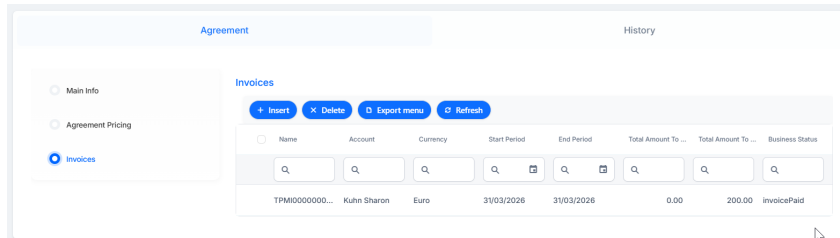
NOTE

You can create as many invoice details as needed for an invoice, but you can create only one invoice detail for the combination between a contract, a commission and a detail type. Distribution Management prevents you to create duplicate invoice detail records containing the same combination of Contract, Commission, and Detail Type values.

Creating Invoices Using the Agreement Page

Follow these steps to create new third-party invoice records directly within the **Agreement** page of an approved agreement record:

1. Open an agreement in **Approved** status and scroll to its **Invoices** section. Here you can insert, delete or export invoices for the selected agreement.



The section displays information about the invoice: name, third-party, currency, start and end period, total amount to pay to the third-party and to recover by your institution, and record's business status.

2. Click **Insert** to display the **Add Invoice** page:
3. Follow the steps [described above](#), starting with **step 2**.

Viewing Invoice Detail Records

The commissions are automatically calculated as invoice details. These invoice details are automatically added to the invoices when the system creates invoices, or you can add them manually to invoices.

To view the list of invoice detail records that are not associated to an invoice yet, follow these steps:

1. In FintechOS Portal, click the main menu icon > **Distribution Management > Distribution Management Operations > Unallocated Invoice Details**.
2. Double-click a record to open it for viewing. You can't perform any other operation on an invoice detail record.
3. On the newly displayed **Invoice Detail** page, view the following information about a calculated commission:

- **Invoice Detail Name** - displayed above the page, it contains the name of the invoice detail.
- **Invoice** - it should contain the invoice number to which it was allocated. Since this is an unallocated invoice detail record, it is empty.
- **Agreement** - the number of the agreement on which this record is based.
- **Detail Type** - the type of payment to be performed: **Payment In** for payments from the third-party's settlement account, or **Payment Out** for payments into the third-party's settlement account.
- **Commission** - the commission used for the calculation.
- **Exchange Rate** - the rate used for calculation.
- **Currency** - the currency code of the calculated commission.
- **Value** - the calculated value of the commission, to be applied to the invoice.
- **Equivalent Value** - the value expressed in the invoice's currency.

Managing Automatic Invoice Payments

Distribution Management identifies the invoices that are not paid on the day of payment specified in the agreement, using the **Charge Not Paid Invoices** scheduled job. The system can automatically generate and process the payments for the invoices, depending on the **Payment Mode** selected at the [invoice level](#).

The method of operating these payments depends on the setting of the [PaymentIsNet](#) system parameter. The value of the parameter specifies whether the system should generate one or two bank account transactions (only for Distribution Management for Banking) and payments for a [third-party agreement invoice](#) when the invoice's status is changed from **Approved** to **Not Paid**.

PaymentsNet = False

If `ThirdPartyPaymentIsNet = False`, when the invoice's status is changed from **Approved** to **Not Paid**, Distribution Management generates one or two bank account transactions with the corresponding payments, as follows:

- If `Total Amount To Recover > 0`, one bank account transaction is generated, with **source** account = `Settlement Account` and **destination** account = `Reconciliation Account` with the value of `Total Amount To Recover`, and with status `Approved`. A payment is generated.
- If `Total Amount To Pay > 0`, another bank account transaction with **source** account = `Reconciliation Account` and **destination** account = `Settlement Account` with the value of `Total Amount To Pay`, and with status `Approved`. A second payment is generated.

NOTE

When `Total Amount To Pay > 0`, `Total Amount To Recover > 0`, and both [payments' statuses](#) become **Allocated**, the invoice's status becomes **Paid**.

PaymentsNet = True

If `ThirdPartyPaymentIsNet = True`, when the invoice's status is changed from **Approved** to **Not Paid**, Distribution Management calculates the difference between `Total Amount To Recover` and `Total Amount To Pay`. Only one bank account transaction is generated and only one payment, representing the non-zero value between the `Total Amount To Recover` and the `Total Amount To Pay`, as follows:

- If `Total Amount To Recover - Total Amount To Pay > 0`, a new bank account transaction is generated with **source** account = `Settlement Account` and **destination** account = `Reconciliation Account`, and a payment is generated for the invoice.

- If $\text{Total Amount To Recover} - \text{Total Amount To Pay} = 0$, a bank account transaction is generated, and the transaction's status changes to **Paid**.
- If $\text{Total Amount To Recover} - \text{Total Amount To Pay} < 0$, a new bank account transaction is generated with **source** account = Reconciliation Account and **destination** account = Settlement Account, and a payment is generated for the invoice.

NOTE

When the [payment's status](#) becomes **Allocated**, the invoice's status becomes **Paid**.

After automatically creating the payment records, Distribution Management displays them for each invoice in the **Invoice** page's **Payments** section:

Each payment record in the list displays information about the payment number, payer's bank account name, transaction date, currency, payment amount, allocated amount, remaining payment amount, total payment amount, and status.

In the **Payments** section, you can search and open for [viewing existing payment records](#), or delete payments in **Draft** status.

NOTE

You can't edit any of the fields of a payment.
Any payment with a status different than **Draft** cannot be deleted

Viewing Third-Party Invoice Payments

To view a third-party invoice payment record, follow these steps:

1. On the **Invoice** page's **Payments** section, double-click the desired payment. The **Payment** page is displayed.

The screenshot shows a web interface for editing a payment. The form is organized into a grid of input fields. On the left, there is a sidebar with two tabs: 'Main Info' (selected) and 'Invoice Details'. The main content area contains the following fields:

- Payment No**: An empty text input field.
- Customer**: A dropdown menu showing 'Goldner Travon' with a blue arrow icon.
- Payer Name**: A text input field containing 'FIN000019220'.
- Transaction Date**: A date input field containing '31/03/2026'.
- Currency**: A dropdown menu showing 'EUR' with a blue arrow icon.
- Bank Reference**: An empty text input field.
- Payment Amount**: A text input field containing '200'.
- Bank Charge**: A text input field containing '0'.
- Total Payment Amount**: A text input field containing '200'.
- Unallocated Amount**: A text input field containing '0'.
- Allocated Amount**: A text input field containing '200'.
- IBAN**: An empty text input field.
- Comments**: A text input field containing 'Third Party Invoice'.

2. View the following information about the selected payment:
- **Payment No** - The number of the payment.
 - **Customer** - The name of the third-party entity associated with the payment.
 - **Payer Name** - The number of the payer bank account.
 - **IBAN** - The IBAN of the account where the money is being paid.
 - **Transaction Date** - The date of the payment transaction.
 - **Currency** - The currency of the payment.
 - **Bank Reference** - The bank reference for the payment.
 - **Bank Charge** - The amount charged by the bank for performing this transaction.
 - **Total Payment Amount** - The sum of the payment amount and the bank charge value.
 - **Payment Amount** - The amount of the payment.
 - **Allocated Amount** - The amount that was already allocated as a contract's repayment for a notification for the selected customer.
 - **Unallocated Amount** - The amount that remains to be allocated as a contract's repayment for a notification for the selected customer.
 - **Comments** - Any comments referring to the payment.

Dashboards and Reports

Distribution Management facilitates user interaction with a series of in-built dashboards.

The dashboards can be accessed from the FintechOS Portal's **Home** page.

The following dashboard comes along with your Distribution Management package:

Third-Party Agreements

The **Third-Party Agreements** dashboard displays a list of the third-party agreements created in the system. The list can be filtered on every column. Access records from the lists by double-clicking them.

The **Add Agreement** button facilitates your access to [the Creating Agreements for Third-Parties page](#), where you can create new agreements.

The dashboard also shows a visual of the agreements within the system, displaying a pie-chart that specifies the number of agreements in each status: Draft, Closed, Approved, and Canceled.

