

Manual Exception Handling

Last Modified on 07/28/2026 6:39 am EDT

V1.0

Overview

Some cases may fail to be processed automatically by the system. The system moves such cases to Exception Handling status based on some pre configured rules.

For example, when the annotation type is Overage, the case status is automatically set to Exception Handling.

Manual exception handling is required for the following scenarios where automated processing is insufficient:

- When the system is unable to complete the processing cycle or line items validation.
- When low-quality files result in incomplete or inaccurate data extraction.
- When the Proof of Delivery (POD) contains notations that require human interpretation, such as overages, returns, refusals, or similar exceptions.
- When the fetched packing list data is missing or incorrect.

Also, some cases might need manual review after validation. An Analyst can manually move such cases to Exception Handling status for review. If an Analyst manually changes the case status from Exception or Validated to Exception Handling, then the Analyst must provide a valid reason that is recorded in the audit log.

For cases in Exception Handling status, the case details fields become editable and an Analyst can manually fix the exceptions and submit the case again for processing.

The routing of cases to Exception Handling is based on business rules configured in the system. You can define these configurable rules to handle the exceptions without code changes. These rules defined in the *ManualException-Mapping.yaml* file determine which scenarios automatically classify a case for Exception Handling. When these rules are met, the system routes the case to Exception Handling for human intervention.

Sample YAML settings

ManualException-Mapping.yaml

```
kind: document
metadata:
  name: deduction/v1/documents/manual-exception-mapping
spec:
  - entity: Backup
    tableName: "dbo.RawBackups"
    fields:
      - fieldName: "Item Name"
        type: "System.String"
        columnName: Description
      - fieldName: "InvoiceNumber"
        type: "System.String"
        columnName: InvoiceNumber
      - fieldName: "PONumber"
        type: "System.String"
        columnName: PoNumber
      - fieldName: "SKU"
        type: "System.String"
        columnName: Sku
        viewColumnName: BckSKU
      - fieldName: "Backup Billed Qty"
        type: "System.Decimal"
```

```

    columnName: InvoicedQuantity
    viewColumnName: BckInvoicedQuantity
- fieldName: "Backup Price/Qty"
  type: "System.Decimal"
  columnName: DeductedPricePerQuantityAmount
  viewColumnName: BckDeductedPricePerQuantity
- fieldName: "Customer SKU"
  type: "System.String"
  columnName: CustomerSku
  viewColumnName: BckCustomerSku
- fieldName: "Deduct Amt"
  type: "System.Decimal"
  columnName: DeductedLineAmount
  viewColumnName: BckDeductedLineAmount
- fieldName: "Deducted Qty"
  type: "System.Decimal"
  columnName: DeductedQuantity
  viewColumnName: BckDeductedQuantity
- entity: Invoice
  tableName: "dbo.RawInvoices"
  fields:
    - fieldName: "InvoiceNumber"
      type: "System.String"
      columnName: InvoiceNumber
    - fieldName: "PONumber"
      type: "System.String"
      columnName: PoNumber
    - fieldName: "SKU"
      type: "System.String"
      columnName: Sku
    - fieldName: "Gross Price"
      type: "System.Decimal"
      columnName: GrossPriceAmount
      viewColumnName: InvGrossPriceAmount
    - fieldName: "Invoice Price/Qty"
      type: "System.Decimal"
      columnName: GrossPricePerQuantityAmount
      viewColumnName: InvGrossPricePerQuantityAmount
    - fieldName: "Invoice Qty"
      type: "System.Decimal"
      columnName: BilledQuantity
      viewColumnName: InvBilledQuantity
    - fieldName: "Invoice QTY (Each)"
      type: "System.Decimal"
      columnName: InvoiceQtyEach
      viewColumnName: InvInvoiceQtyEach
    - fieldName: "Item Name"
      type: "System.String"
      columnName: Description
      viewColumnName: InvItemDescription
    - fieldName: "Net Price"
      type: "System.Decimal"
      columnName: NetPriceAmount
      viewColumnName: InvNetPriceAmount
- entity: Pod
  tableName: "dbo.RawPodNotation"
  fields:
    - fieldName: "SKU"
      type: "System.String"
      columnName: ItemIdentifier
    - fieldName: "Damaged"
      type: "System.Decimal"
      columnName: Quantity
      viewColumnName: PodTotalDamage

```

```
viewColumnName: PodTotalDamage
- fieldName: Overage
  type: "System.Decimal"
  columnName: Quantity
  viewColumnName: PodTotalOverage
- fieldName: Shortage
  type: "System.Decimal"
  columnName: Quantity
  viewColumnName: PodTotalShortage
- entity: LineItemValidation
  tableName: "dbo.LineItemValidation"
  fields:
    - fieldName: "InvoiceNumber"
      type: "System.String"
      columnName: InvoiceNumber
    - fieldName: "PONumber"
      type: "System.String"
      columnName: PoNumber
    - fieldName: "SKU"
      type: "System.String"
      columnName: Sku
    - fieldName: Confirm
      type: "System.Boolean"
      columnName: Confirm
    - fieldName: "Denominator"
      type: "System.Decimal"
      columnName: Denominator
      viewColumnName: Denominator
    - fieldName: "Invalid Amount"
      type: "System.Decimal"
      columnName: InvalidAmount
      viewColumnName: InvalidAmount
    - fieldName: "Invalid Quantity"
      type: "System.Decimal"
      columnName: InvalidQuantity
      viewColumnName: InvalidQuantity
    - fieldName: "Invalid Reason"
      type: "System.String"
      columnName: InvalidReason
      viewColumnName: InvalidReason
    - fieldName: "Valid Amount"
      type: "System.Decimal"
      columnName: ValidAmount
      viewColumnName: ValidAmount
    - fieldName: "Valid Quantity"
      type: "System.Decimal"
      columnName: ValidQuantity
      viewColumnName: ValidQuantity
    - fieldName: "Validation Status"
      type: "System.String"
      columnName: ValidationStatus
      viewColumnName: ValidationStatus
```

Parameter	Description
-----------	-------------

Parameter	Description
entity	The system/resource to be mapped for exception handling fields. For example: • Invoice • Backup • PoD • LineItemValidation • Packing list
tableName	The specific table in DB to map the fields.
fields	The fields to map for exception handling.

manual-exception-row-enrichers.yaml

```
# Purpose:
# Turn a SINGLE input line that carries multiple "notation" quantities
# (e.g. one item with Shortage=10, Damaged=1, Overage=30) into MULTIPLE
# output rows -- one per listed field. Required for tables like
# dbo.RawPodNotation where each notation is its own row.
# Column rule interpretation:
# - if `value` is set -> emit that literal
# - otherwise -> emit the expanded field name (optionally
# translated via `map` ).
# -----
kind: document
metadata:
  name: deduction/v1/documents/manual-exception-row-enrichers
spec:
  - entity: Pod
    tableName: "dbo.RawPodNotation"
    expandFields:
      - Damaged
      - Overage
      - Shortage
    columns:
      - column: NotationType
        map:
          Damaged: Damaged
      - column: QuantityType
        value: "N/A"
```

Parameter	Description
entity	The system/resource to be mapped for exception handling fields.
tableName	The specific table in DB to map the fields.
expandFields	The extended fields to map for exception handling.
columns	The columns to map for exception handling

