|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | | |
| **Technical Specification**  WKF014  Payment Approval Workflow | | |
|  | | |
| Prepared for: | Madison Group Enterprises | |
| Version: | Version 0.3 | |
| Version Release Date: | 09/03/2024 | |
| Owner of Document: | Kristina Verco | |
|  |  | |
|  |  | |

**Version Control**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Section** | **Description** | **Changed By** |
| V0.1 | 20/02/2024 |  | First draft | Kristina Verco |
| V0.2 | 28/02/2024 |  | Only approved lines sh | Kristina Verco |
| V0.3 | 09/03/2024 |  | Changed M3 Program Name to use as name specified was not available for use and adding additional index fields required and more information of type of index required. | Theo Tataroglou |
|  |  |  |  |  |
|  |  |  |  |  |

**Stakeholders**

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | **Company** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Related Documents**

|  |  |  |
| --- | --- | --- |
| **Document Name** | **Location** | **Owner of the document** |
| WKF014- Payment Approval\_Madison v1.1.docx | Sharepoint | Bharatha Gunathilaka |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Table of Contents**

1. Solution Summary 4

1.1 Assumptions and Constraints 4

2. Technical specification 4

2.1 Process Flow 4

2.2 ION Workflow 5

2.2.1 Payment\_Proposal\_Approval Trigger 5

2.2.2 Payment\_Proposal\_Approval Workflow 5

2.3 XtendM3 Database Definition 6

2.3.1 EXTAPP – Approval Payment Proposal 6

2.4 XtendM3 APIs 7

2.4.1 EXT500MI.Add 7

2.4.2 EXT500MI.Delete 8

2.4.3 EXT500MI.Update 8

2.4.4 EXT500MI.Get 9

2.4.5 EXT500MI.GetDailyAmount 10

2.5 H5 Script 10

2.5.1 APS130/A Panel 10

2.5.2 APS130/B Panel 11

2.6 Error Handling 11

# Solution Summary

The Payment Approval workflow supports a one-level DoA approval process with daily $ approval limits for groups. An XtendM3 table will be used to store the total approved amount for each group in local currency and the workflow will determine when the maximum daily approval limit has been reached and escalation is required.

# Assumptions and Constraints

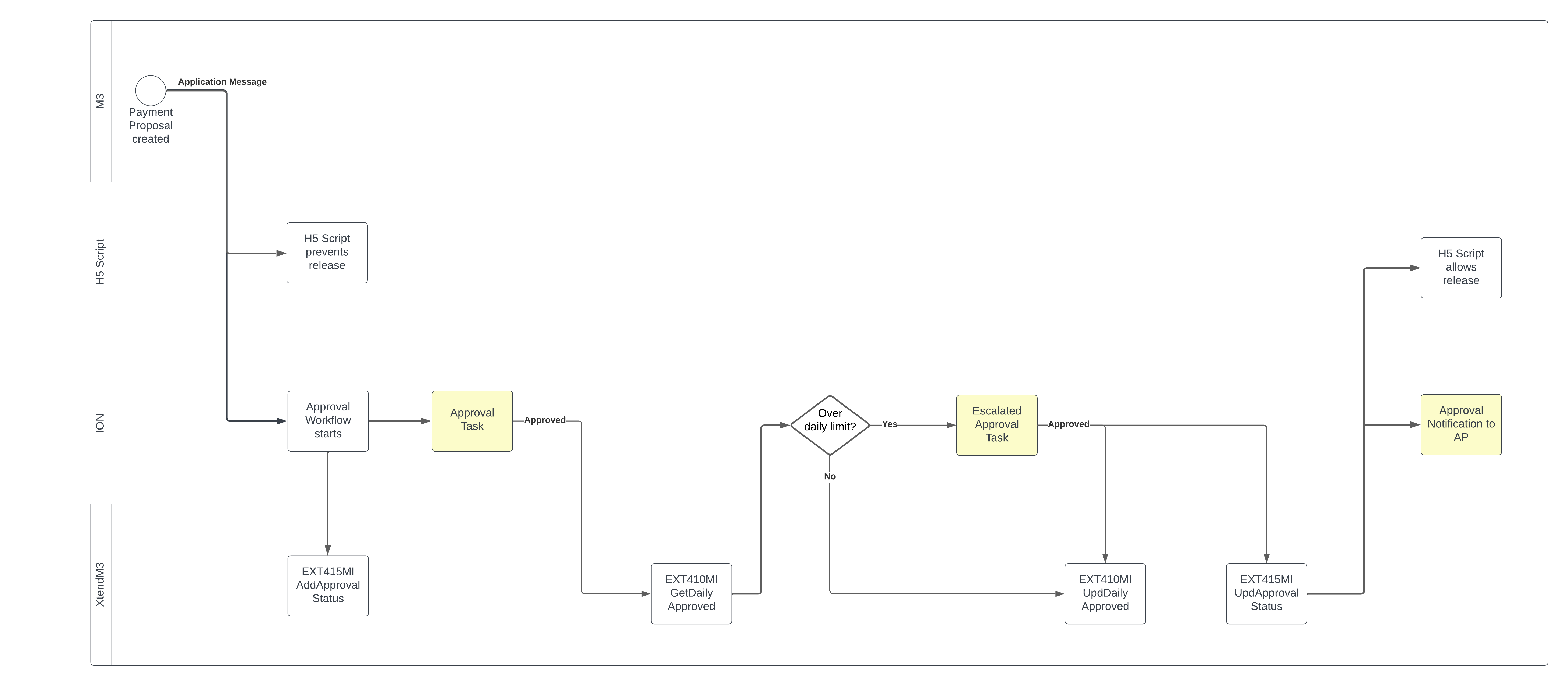
The following assumptions have been made in this design:

* Only a single escalation is required. There is no need for a second check of the daily limit and second escalation.
* Approval status and amount will be stored in an XtendM3 table.
* It is assumed that if the payment proposal workflow needs to be restarted, it will be done manually from the Start Workflows widget by an authorised user.
* Changes to the payment proposal will not automatically trigger a new workflow. It is expected that this will be managed within the workflow tasks by AP.
* All daily approval limits will be stored in local currency.
* Approval limits will apply to local timezone dates.
* Approver Groups will be configured as Groups in IFS.

# Technical specification

## Process Flow

The following Process Flow diagram describes the relationship between technical components.



## ION Workflow

### Payment\_Proposal\_Approval Trigger

The Payment\_Proposal\_Approval ION workflow will be triggered on Payment Proposal creation by the ‘Approval of supplier payment proposal is required’ application message 061 in CRS424.

A screenshot of a computer

Description automatically generated

### Payment\_Proposal\_Approval Workflow

The workflow processes the approval with the following logic:

* Convert the Payment Proposal amount to local AUD currency
* On workflow initiation, call EXT500MI.Add with key data and:
  + ASTS – ‘Under Approval’
  + APPR – ‘’
  + GRPI – <Approving Group based on DoA rules>
  + APAM - $0

If an entry already exists for this Payment Proposal, reset the data with EXT500MI.Update and the values above.

* Send approval task to the Approving Group.
* On Approve action, call EXT500MI.GetDailyAmount to retrieve the approved amount for today’s date.
  + If APAM (Approved Amount) + Payment Proposal local currency amount is above the DoA limit for the group:
    - Get next level DoA group with correct limit
    - Call EXT500MI.Update with key data and:
      * ASTS – ‘Under Approval’
      * APPR – ‘’
      * GRPI – <Next Level Group ID based on DoA rules>
      * APAM - $0
    - Send escalation task to Next Level DoA group.
    - On Approve action:
      * Call EXT500MI.Update with key data and:
        + ASTS – ‘Approved’
        + APPR – <Approving User ID>
        + GRPI – <Approving Group ID>
        + APAM – <Payment Proposal local currency amount>
    - On Decline action:
      * Send task to AP for cancellation or resubmit to current DoA Group
  + If APAM (Approved Amount) + Payment Proposal local currency amount is within the DoA limit for the group:
    - Call EXT500MI.Update with key data and:
      * ASTS – ‘Approved’
      * APPR – <Approving User ID>
      * GRPI – <Approving Group ID>
      * APAM – <Payment Proposal local currency amount>
* On Decline action:
  + Send task to AP for cancellation or resubmit to current DoA Group

# XtendM3 Database Definition

An XtendM3 table will be used by the workflow to store approval status, approval dates and local currency amounts.

### EXTAPP – Approval Payment Proposal

The Approval Payment Proposal table stores approval status details.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column Name | Description | Type | Length | Number of Decimals |
| EXCONO | Company | DECIMAL | 3 | 0 |
| EXDIVI | Division | STRING | 3 |  |
| EXPRPN | Payment Proposal | DECIMAL | 10 | 0 |
| EXPYON | Payment Order | DECIMAL | 5 | 0 |
| EXINYR | Invoice Year | DECIMAL | 4 |  |
| EXASTS | Approval Status | STRING | 30 |  |
| EXAPPR | Approver | STRING | 10 |  |
| EXGRPI | Group ID | STRING | 30 |  |
| EXDATE | Approval Date (in YYYYMMDD) | DECIMAL | 8 | 0 |
| EXAPAM | Approved Amount | DECIMAL | 15 | 2 |
| EXRGDT | Entry Date | DECIMAL | 8 | 0 |
| EXRGTM | Entry Time | DECIMAL | 6 | 0 |
| EXLMDT | Change Date | DECIMAL | 8 | 0 |
| EXCHNO | Change Number | DECIMAL | 3 | 0 |
| EXCHID | Change ID | STRING | 10 |  |

Indexes:

|  |  |
| --- | --- |
|  | Columns |
| 00 *(Unique)* | EXCONO, EXDIVI, EXPRPN, EXPYON *(All Ascending)* |
| 10 | EXCONO, EXASTS, EXDIVI, EXPRPN, EXPYON *(All Ascending)* |
| 20 | EXCONO, EXGRPI, EXDATE, EXDIVI, EXPRPN, EXPYON, EXAPAM, EXASTS *(All Ascending)* |

# XtendM3 APIs

# EXT500MI.Add

Input (inData):

* CONO
* DIVI
* PRPN
* PYON
* ASTS
* APPR
* GRPI
* APAM

Output (outData):

N/A

Add processing:

* Validate that ASTS matches one of the following values:
  + ‘Sent for Approval’
  + ‘Under Approval’
  + ‘Under Cancellation’
  + ‘Approved’
  + ‘Declined’
  + ‘Rejected’
  + ‘Cancelled’
* Validate that APPR is either blank or is a valid user id in table CMNUSR with:
  + JUCONO = 0
  + JUDIVI = ‘’
  + JUUSID = APPR
* Validate that an entry does not already exist in EXTAPP with this key. If it does, return mi.error, “An approval entry already exists for this transaction”.
* Validate that the key passed to the API is valid in the FPSUGH table.  
  If a valid table row is not found, return mi.error, “Transaction is invalid”.
* Add a row to EXTAPP with the inData provided.

# EXT500MI.Delete

Input (inData):

* CONO
* DIVI
* PRPN
* PYON

Output (outData):

N/A

Delete processing:

* Validate an entry exists in EXTAPP with this key. If not found, return mi.error, “An approval entry does not exist for this transaction”.
* Delete the row on EXTAPP.

# EXT500MI.Update

Input (inData):

* CONO
* DIVI
* PRPN
* PYON
* ASTS
* APPR
* GRPI
* APAM

Output (outData):

N/A

Update processing:

* Validate that ASTS matches one of the following values:
  + ‘Sent for Approval’
  + ‘Under Approval’
  + ‘Under Cancellation’
  + ‘Approved’
  + ‘Declined’
  + ‘Rejected’
  + ‘Cancelled’
* Validate that APPR is either blank or is a valid user id in table CMNUSR with:
  + JUCONO = 0
  + JUDIVI = ‘’
  + JUUSID = APPR
* Validate an entry exists in EXTAPP with this key. If not found, return mi.error, “An approval entry does not exist for this transaction”.
* Update the row on EXTAPP with the APPR, ASTS, GRPI, APAM provided. If ASTS=’Approved’, update DATE with the current date in local timezone.

# EXT500MI.Get

Input (inData):

* CONO
* DIVI
* PRPN
* PYON

Output (outData):

* CONO
* DIVI
* PRPN
* PYON
* ASTS
* APPR
* GRPI
* APAM
* DATE

Get processing:

* Validate an entry exists in EXTAPP with this key. If not found, return mi.error, “An approval entry does not exist for this transaction”.
* Output the row details from EXTAPP.

# EXT500MI.GetDailyAmount

GetDailyAmount returns the total approved amount for the current date and group id.

Input (inData):

* CONO
* GRPI

Output (outData):

* CONO
* GRPI
* DATE
* APAM

GetDailyAmount processing:

* Validate that CONO is valid in CMNCMP.
* Validate that GRPI is not blank or return mi.error, “An Approval Group must be supplied”.
* Populate DATE with the current date in local timezone.
* Sum the APAM total for every row in EXTAPP with CONO, GRPI and DATE and where ASTS = ‘Approved’.

## H5 Script

In APS130/A and APS130/B panels, Accounts Payable users can release a Payment Proposal with option 9-Confirm. A new script will validate whether the proposal has been approved before allowing the proposal to be confirmed.

# APS130/A Panel

A close up of a screen

Description automatically generated

* Retrieve Division, Payment Proposal and Payment Order from the display.  
  If Division is not available, retrieve session Division value.
* Call EXT500MI.Get to retrieve the Payment Proposal’s approval status. If the API does not return ASTS=’Approved’, or if it returns an error, show dialog message ‘Payment Proposal cannot be confirmed until approved. Current approval status is <ASTS>’.

# APS130/B Panel

A screenshot of a computer

Description automatically generated

* Retrieve Division, Payment Proposal and Payment Order from the selected row(s).  
  If Division is not available, retrieve session Division value.   
  NB This script should work in both Sorting order 1 and 2.
* Call EXT500MI.Get to retrieve the Payment Proposal’s approval status. If the API does not return ASTS=’Approved’, or if it returns an error, show dialog message ‘Payment Proposal cannot be confirmed until approved. Current approval status is <ASTS>’.

## Error Handling

Errors in the workflow such as no valid approvers or API errors will trigger a message to group IT\_ERRORS for follow up.