**PEG**

**Is Optional:**

The "Skip" option is only available for transactions of the "Make" type.   
When a "Make" transaction has the "Is optional" enabled, a "Skip" / "Next" button will appear on the form.   
Clicking on that button will allow you to proceed to the next task.

 This functionality applies to both parallel and sequential tasks. It also works in the same way for return processes.

**PEG notifications:**

As mentioned in the earlier release notes, there are three **Application variables** that need to be created and maintained for processing PEG reminders, escalations, and notifications.

1) **AxScriptsAPIURL**: This variable holds the ASBScriptsREST API URL, which is used to process the reminders, escalations, and notifications. It should be set to the correct URL during promotion or deployment.

**Syntax:**   
{BaseScriptsURL}/ASBScriptRest.dll/datasnap/rest/TASBScriptRest/scriptsapi

**Example:**   
<https://app.buzzily.com/UATScripts/ASBScriptRest.dll/datasnap/rest/TASBScriptRest/scriptsapi>

2) **AxRMQAPIURL**: This variable holds the ARM API URL, which is used to push reminder, escalation, and notification-related messages to the RMQ server when performing actions on PEG forms.

**Example:**   
<http://20.244.123.19/ARMTest/api/v1/ARMPushToQueue>  
(Note: This should point to the location where the ARM APIs are hosted.)

3) **AxPEGMailFrom**: This variable holds the email address that will be used as the "From" email address for PEG-related email notifications (reminders, escalations, and notifications).

**Example:**   
hcm.support@agile-labs.com

Please ensure that these variables are correctly set and maintained for proper functioning of PEG reminders, escalations, and notifications.

**Skip Level:**

The term "SkipLevel" refers to the **user's reporting manager's reporting manager**. It means that if User A, User B, and User C are involved in a process.

Where User A's RM User B.

            User B's RM User C.

And User A initiates the process and the "AssignTo" field as "SkipLevel" for approval, the process task will be assigned to User C, who is User B's reporting manager.

In this, "SkipLevel" indicates the level above the direct reporting manager. It allows for skipping one level in the reporting hierarchy and assigning the task to the reporting manager's reporting manager instead.

**Initiator skip level** - Initiator's reporting manager's reporting manager.  
**User skip level** - Current users reporting manager's reporting manager.

**Send in PEG:**

This is an option that can be enabled optionally in approve and check tasks. User will have an option to enable “Send” option. This task can be sent by this user to None, Any user or one or more actors. If “none” is selected, this task cannot be sent to other users.

When a task is sent to another user, the receiver can send to another user or act on it. When a user sends to another user, the record will be moved from the pending list of the sender and moved to the receiver.

The receiver becomes the owner of the task and can approve, reject, return, mark as checked or may send to another user. The data will continue in the work flow when the receiver acts on it.

When a task is sent, the sender can optionally enter a comment.

**Auto Approve Task:**

"Auto Approve" checkbox in the Approve type Tstruct. When this checkbox is selected, the auto approval process will be triggered during the PEG process.

If auto approve checkbox is selected then following fields and DC’s will be hidden,

**Fields**

Assign to, Allow user to Send to\*, Auto approve for orphan records, Is return allowed, Consider only business dates for Escalations/Reminders, Return notifications, Reject notifications

**DC’s**

Reminders, Escalation and Approval comments.

**Option to Configure Business Date Calculation:**

In this release, an option is given in the task level to configure the calculation of the business date. If this option is checked, Axpert will use AxPEGGetBusinessDate function to calculate the business date. If the option is unchecked, then Axpert will use our existing logic (no business day logic will be followed).

Note that solution developers can write this function according to their project requirement. The function name must be **AxPEGGetBusinessDate**, and the parameters must be in the same order as given below.

function AxPEGGetBusinessDate(  
  TaskId Varchar(50),   
  StartDate timestamp,   
  ElapsedDays integer,   
  ElapsedHours integer,   
  ElapsedMinutes integer  
) returns timestamp

**Amendments:**

This functionality is applicable only to edit transactions and not for new transactions. The changed data will not be directly pushed to the Transaction table. Instead, the changes will be sent for approval if an amendment flow is defined in the PEG definition. Only upon final approval, the amendments will be pushed to the actual Transaction table.

Until the data is approved, it will be kept in the AxAmend table. When the user loads the record, the amendments will be displayed in a status bar.

**Design:**

Introduce a check box in process master as “Amendment flow”

The rest of the definition remains the same as the PEG process.

At run time, when user submits to a form in edit mode, search in process definition table for an amendment flow for the tstruct. If exists, then call SaveAmend service. This service will save changes into AxAmend table and post to AxActiveTasks table.

On approval in the final step, make a call to AcceptAmend REST service. The amend no will be passed to the service. This will load data, apply the changes to the fields from the AxAmend table and save the transaction.

On discard, in any step, make a call to DiscardAmend REST service. This will set the status accordingly in the AxAmend table for all the records for the given amend no.

In tstruct run time user interface – if a transaction that is loaded in edit mode, check if the AxAmend has records that are waiting approval. If it has, show a line below the title as “Amendments done by <user name> on <date time> are awaiting approval. Click here to view amendment details.” On click of the link, expand this div to show all the changes. If this user has authority to act on these amendments, show the accept, discard and return buttons at the bottom of this div. On click make respective service calls & reload the record.

**Condition in Make Task:**

**Allow to edit prior forms**

**Partial Edit**

**Full Form Edit**

**Auto approve for orphan records**

**Allow initiator to approve**