Oracle Fusion Cloud Global Payroll

How do I set up payroll task iterations?

FA-Latest

Oracle Fusion Cloud Global Payroll
How do I set up payroll task iterations?

FA-Latest

G31014-02

Author: Susan Saha

Contents

Get Help

There are a number of ways to learn more about your product and interact with Oracle and other users.

Get Help in the Applications

Some application pages have help icons to give you access to contextual help. If you don't see any help icons on your page, click your user image or name in the global header and select Show Help Icons. If the page has contextual help, help icons will appear.

Get Training

Increase your knowledge of Oracle Cloud by taking courses at *Oracle University*.

Join Our Community

Use *Cloud Customer Connect* to get information from industry experts at Oracle and in the partner community. You can join forums to connect with other customers, post questions, suggest *ideas* for product enhancements, and watch events.

Share Your Feedback

We welcome your feedback about Oracle Applications user assistance. If you need clarification, find an error, or just want to tell us what you found helpful, we'd like to hear from you.

You can email your feedback to *oracle_fusion_applications_help_ww_grp@oracle.com*.

Thanks for helping us improve our user assistance!

1 Setup Summary

Scope

Use this playbook to set up flow iterations to automatically repeat a task multiple times with a single submission of the flow.

It includes steps to create a flow and include the **Submit Another Task** task to initiate task iterations,. It also details how you can define rules to control the task iterations and define more meaningful task iteration names to identify and monitor the individual iterations easily.

Examples of Using Task Iterations

This topic gives examples of how you can use task iterations to meet a business requirement within your organization.

Scenario 1

Your company doesn't process payments for the QuickPay runs, and hence, before you run the payroll, you must roll back all the QuickPay runs within the payroll period. If you don't roll back the QuickPay runs, those persons processed by the QuickPay runs aren't picked up in the payroll run and hence, don't receive their payments.

Rather than manually roll back the individual QuickPay runs, use the task iterator to dynamically identify the transaction IDs of the QuickPay runs within the specified period and automate the roll back of the QuickPay runs.

Scenario 2

As a payroll administrator, you need to track employee balances of all your employees across all of the payroll statutory units (PSUs) within your organization for year-end reporting.

To meet this requirement, you create a flow pattern and use the Submit Another Task task to automatically generate the Archive End-of-Year (EOY) report for each individual PSU within your organization.

The task iteration process uses the task repeat formula to dynamically identify the PSU IDs and generate the report for the multiple PSUs with a single submission of the flow. The Archive EOY Payroll Results task retrieves employee and employer information and employee balances in a given year for year-end reporting.

2 Create Task Iterations

Task Iterations in a Flow

Use the **Submit Another Task** task to automatically repeat a flow task multiple times with a single submission of the flow.

Define rules to control the task iteration. For example, define a rule to automatically submit a report task for each PSU within your organization.

There are no predefined flows that you can use directly for task iterations. To iterate a task, you must create a flow and include the 'Submit Another Task' task in the flow.

Consider using a standard flow pattern when your task submission rules are based on a static parameter such as TRU and you have a requirement to manage each task individually.

The rule-based nature of the 'Submit Another Task' task enables you to support submission rules based on fluid parameter details. For example, before calculating your regular payroll run, you can automatically roll back all QuickPay runs you have created and submitted for payroll validation purpose. In this instance, the rollback task is automatically resubmitted for all the QuickPay runs in the payroll period.

> **Note:** You can't iterate a manual task. For example, you can't use the Submit Another Task option to iterate a QuickPay task and submit multiple QuickPay runs.

Consider these points before you create a flow and include Submit Another Task to initiate task iterations.

Task Name

Submit Another Task takes the task name as the input parameter and uses the repeat formula to execute multiple iterations of the task.

Task Iteration Name

To differentiate the multiple iterations and easily identify an iteration counter in the Submit Another Task UI, it's recommended that the formula you use for the iterations include a meaningful label name for each iteration.

Use the Task Repeat formula to return an iteration label that includes the task iteration name appended with the parameter on which the iterations are executed.

For example for a flow iteration to roll back Calculate Payroll processed for different payroll relationship groups, you can use the formula to return the iteration names as Roll Back Payroll Relationship Group A, Roll Back Payroll Relationship Group B, and so on. The task name is constant for the task iterations, but the parameter on which the iterations are executed is dynamic.

Task Repeat Formula

The Task Repeat Formula is a prerequisite for the flow and it decides the repetition logic and drives the iteration. When you create the repeat formula, use the formula type of **Task Repeat**.

The input parameters are predefined for an input task. You can add them as flow parameters during flow creation. The formula return values are used to validate the task parameters.

Maximum Repeat Counter

Specify a threshold limit for the maximum number of instances that can be executed for a single submission of the task. If the iteration runs into an error, this parameter prevents the process from getting into an infinite loop. A repeat counter N indicates one parent and N-1 child submissions or iterations.

Repeat Flow Parameter

The repeat flow parameter indicates when the task iteration should stop. A repeat flow instance is submitted only if the repeat flow parameter is set to **Y**.

Execute in Parallel

Execute the submissions in parallel or serial.

- For parallel submissions, the number of threads is taken into consideration. A number of submissions equivalent to the number of threads is submitted in parallel.
- For serial submissions, number of submissions equivalent to the Repeat Counter is executed one after the other.

If you have set the Execute in Parallel parameter to **Yes**, you must specify a value for the Maximum Parallel Threads. This parameter is used for a parallel submission.

For example, if you specify a value of X, during a submission, the application processes X instances of the task initially. The parent task waits for X child submissions to complete, evaluates the fast formula and executes another set of X child submissions to complete. This cycle continues till the Maximum Repeat Counter submissions are completed or the Repeat Flow value is 'N'.

Note: It is recommended to execute the flow with parallel threads, especially for large volumes of data.

Parent Log File

After completion of the flow, use the Task Iterations page from the Process Results Summary and view the ESS Log file to view the details of the input parameters for each job submission. The Output and Log Files section displays one set of flow instance details and parameter values for the parent submission and similar set of values for each subsequent child submission.

- Click on the Process ID on the first row to view the log output details of the flow.
- Click on the Process ID on the second row or any other row below to view the log output of a task iteration or child submission.

Flow Parameters for Task Iterations

Task parameters submit the information required for the task submissions to complete successfully.

Task iteration flows have two sets of parameters:

- Parameters for Submit Another Task
- Parameters for the repeat submission task that's submitted multiple times

Use any of these options as input values for the repeat submission task:

- Task parameters from the flow parameters defined while creating the flow
- Return parameters from the repeat formula

Use any of these options as flow parameter values:

- If the task parameter name, exposed in the UI at the flow-level, matches the Base Flow Parameter Name, then the values are used directly.
- If the task parameter names don't match the Base Flow Parameter Name, the application uses a fast formula function to pass the flow parameter values as the task parameters. This formula function is built into the repeat formula.
  For example, assume that you have defined the flow parameter as 'Effective Date', and the corresponding Base Flow Parameter Name is 'EFFECTIVE DATE'. The application uses formula function `GET_FLOW_PARAM_VALUE ()` and stores the return value of this formula function in the task parameter 'EFFECTIVE DATE'. The formula function is:
  - `*EFFECTIVE DATE=GET_FLOW_PARAM_VALUE` ('Effective Date')

Provide the correct parameter basis when you define the task parameters.

- Some of the static parameters like the Effective Date or the Start Date can have a parameter basis value of 'Bind to Flow'.
- The dynamic parameters like the Payroll Statutory Unit ID are derived from the database tables. Hence you can have a parameter basis value of 'Bind to Flow Task' or 'Context Binding'.
- If you have defined specific names for the flow task parameters, you must ensure that the same names are used in the repeat formula.

Alternately, use the return parameters from the repeat formula as the task parameters. In this case the input parameters for the repeat formula are only the Base Task Name and the Repeat Counter. While calling the formula, the application uses these two parameters to get the context of the job submission. You create the Repeat formula to return input values for the submission task.

The formula output Repeat Flow decides if another job submission is required.

For example, for the Archive End-of-Year Payroll Results process the following parameters are defined as static parameters:

- Effective Date
- Start Date
- Tax Year Date
- Repeat Counter
  The Repeat Counter is a static variable and is maintained by the application. During the iteration process this parameter increments by '1' after every submission.

In this example, these are the dynamic parameters for each submission:

- Payroll Statutory Unit ID

- Repeat Flow

However, you can also define the Tax Year as a dynamic parameter, so that you can generate the report for various years.

Automate Roll Back of Multiple QuickPay Runs

In this example, you use the **Submit Another Task** task to automatically roll back multipe QuickPay runs with a single flow submission.

Before you begin to automate the rollback of the QuickPay runs, complete these tasks:

1. Create a value set to derive the transaction ID of the QuickPays already processed. The flow iterator uses the transaction IDs for the multiple iterations.
2. Create the task iterator fast formula of type 'Task Repeat'.
3. Create a user-defined flow and include the 'Submit Another Task' task for iterating the task.

Create a Value Set

Let's look at the steps to create a value set that returns all the QuickPay transaction IDs.

1. Select the **Payroll Value Sets** task in the Setup and Maintenance work area.
2. On the Value Sets page, click **Create** and enter these values.

Field	Value
Value Set Code	ROLLBACKQPAY_VS
Description	QuickPay Rollback
Module	Payroll Flows
Validation Type	Table
Value Data Type	Number
FROM Clause	pay_payroll_actions pa, pay_requests pr
Value Column Name	pa.payroll_action_id
Value Column Type	VARCHAR2
Value Column Length	200

Field	Value
ID Column Name	pa.PAYROLL_ACTION_ID
ID Column Type	Number
ID Column Length	18

WHERE Clause	`pa.pay_request_id(+) = pr.pay_request_id`
	`and pr.pay_request_id =(`
	`select pay_request_id from (`
	`select t1.pay_request_id,rownum cnt from(`
	`select pr.pay_request_id,pfi.instance_name`
	`,nvl(pa.payroll_action_id,-1) payroll_action_id`
	`from pay_flow_instances pfi,`
	`pay_flows pf,`
	`pay_requests pr,`
	`pay_payroll_actions pa`
	`where pfi.base_flow_id =pf.base_flow_id`
	`and pf.base_flow_name IN ('QUICK_PAY',`
	`'QUICK_PAY_ONLY',`
	`'QUICK_PAY_SIMPLIFIED')`
	`AND pr.flow_instance_id=pfi.flow_instance_id`
	`AND pa.pay_request_id(+) = pr.pay_request_id`
	`and pfi.instance_name like 'QuickPay_XXXX,'`
	`and pr.call_type='FLOW_TASK_INSTANCE'`
	`order by pr.pay_request_id desc`
	`)t1`
	`)`
	`where cnt=:{PARAMETER.COUNT}`
	`)`

ORDER BY Clause	

Note:

- The SQL in this table is a sample to demonstrate how you can use a value set to retrieve transactions for the QuickPay rollback iterations.
- The Where Clause below the SQL is to restrict the rollback iterations to process only those flow instances beginning with the specified name. For example, 'QuickPay_XXXX' is used as the flow name in this example. 'XXXX' could represent day and month when the QuickPay runs are processed.

3. Optionally, to secure the value set, select the Data Security check box and provide the Data Security Resource Name.

Note: You can enable data security only if the value set is based on a single table or view.

4. Click **Save**.

After you create the value set, perform these actions to check if the value set query is correct and working as expected:

1. Set up a flow with just this value set.
2. Verify that the flow generates the list of transaction IDs of the processed QuickPays.
3. Verify that the QuickPay IDs are captured as a flow parameter value on the flow submission page.
4. Click **Continue**.
5. Enter this formula text details in the Formula Text Section.

```
1. FORMULA NAME: QP_ROLLBACK
2. FORMULA TYPE: Task Repeat
3. DESCRIPTION: Formula to roll back the QuickPays
4. Formula Results: Iterates the Roll Back QuickPay process
5. /* Inputs */
6. INPUTS ARE FLOW_INSTANCE_ID, REPEAT_COUNTER, BASE_TASK_NAME (text)
7. DEFAULT FOR REPEAT_COUNTER IS 1
8. DEFAULT FOR FLOW_INSTANCE ID IS 1
9. DEFAULT FOR PAYROLL_PROCESS IS 1
10. COUNTER = to_char(REPEAT_COUNTER)
11. PAYACTION_ID = GET_VALUE_SET ('ROLLBACKQPAY_VS' ,'|=COUNT='|| COUNTER)
12. BASE_TASK_NAME = 'JAVA_API_ROLLBACK'
13. EXECUTION_MODE = 'SUBMIT'
14. EFFECTIVE_DATE = trunc(to_date(GLOBAL_PAY_INTERFACE_EXTRACTION_DATE,'YYYY-MM-DD'))
15. PAYROLL_PROCESS = PAY ACTION_ID
16. REPEATFLAG = 'Y'
17. /*Results*/
18. RETURN REPEAT_COUNTER, REPEAT FLAG, BASE_TASK_NAME, EXECUTION_MODE, EFFECTIVE_DATE, PAYACTION_ID,
 PAYROLL_PROCESS
19. /* End Formula Text */
```

6. Click **Compile**.
7. Click **Save**.

Create the Task Repeat Formula

The Task Repeat Formula calls the value set you previously defined to retrieve the list of QuickPay transaction IDs for the QuickPays already processed. The formula return values are used to validate the task parameters for the Rollback Process in order to roll back in each iteration the current QuickPay in the iteration logic.

Use the text editor to create a fast formula and return the values required to run the Roll Back QuickPay process.

Complete these steps to create a repeat formula.

1. Use the **Fast Formulas** task in the Setup and Maintenance work area.
2. On the Fast Formulas page, click **Create** to create a formula.
3. On the Create Fast Formula page, complete these fields.

Field	Value
Formula Name	QP_Rollback
Formula Type	Task Repeat
Description	Formula to roll back QuickPay

Field	Value
Effective Start Date	Don't enter a value. It is derived by the task repeat formula.
Legislative Data Group	This formula isn't specific to any legislative data group (LDG) and hence you can leave this field blank.
Type of Editor	Text

Create a User-Defined Flow Pattern

Complete these steps to create a flow pattern.

1. On the Home page, navigate to **Payroll** in **My Client Groups** and select the **Payroll Flow Patterns** task.
2. Click **Create** to create a flow pattern. You can also search for and select an existing flow pattern to copy.
3. Leave the Legislative Data Group field blank and click **Continue**.
4. On the Basic Information page, complete these basic flow information fields.

Field	Value
Flow Pattern Name	Roll Back QuickPay
Description	Enter a description for the flow.
LDG Required	No
Activities to Include	Select Calculate.

5. Select **Submit Another Task** to include it in the flow pattern.
6. Select an **Owner Type** and **Owner** for the flow.
7. On the Task Sequence page, the sequence should appear as given here.

Task	Following Task
Start Task	Submit Another Task
Submit Another Task	End Task

8. On the Parameters page, use the **Add** icon and add the requisite flow parameters to complete the flow pattern.
9. Before you add the parameters, click **Next** and review the task parameters. The parameters associated with Submit Another Task are automatically available. Ensure that the parameters are defined as given in this table.

Name	Execution Mode	Data Type	Parameter Basis	Basis Value
Execute in Parallel	Submit	Text	Bind to Flow	Execute in Parallel
Legislative Data group	Submit	Long	Context Binding	Legislative Data group
Task Name	Submit	Text	Bind to Flow	Task Name
Task Repeat Formula	Submit	Text	Bind to Flow	Task Repeat Formula
Maximum Repeat Counter	Submit	Long	Bind to Flow	Maximum Repeat Counter
Request	Submit	Long	Context Binding	Request
Maximum Parallel Threads	Submit	Number	Bind to Flow	Maximum Parallel Threads

10. The flow parameters are used to submit and complete the tasks in the flow pattern, or as a basis for deriving values to submit the remaining tasks in the flow pattern.
11. If necessary update the parameters. After you have completed the requisite flow parameters, click **Next**.
12. Review the resulting checklist for the flow pattern.
13. Click **Submit**.

Submit the Flow

Complete these steps to submit the newly created flow pattern.

1. Select the **Submit a Flow** task from Quick Actions or under **Payroll** in **My Client Groups** on the Home page.
2. Search for **Roll Back QuickPay** flow and click **Next**.
3. Enter these parameters.

Field	Value
Payroll Flow	Enter a suitable name for the flow instance, for example Roll Back QuickPay_0105.
Schedule	Select **As soon as possible** or select **Using a schedule** and enter the schedule details.
Task Name	Roll Back Process
Task Repeat Formula	QP_Rollback

Field	Value
Maximum Repeat Counter	Maximum number of submissions or the number of iterations by the flow. **Note:** A repeat counter N indicates one parent and N-1 child submissions.
Execute in Parallel	Decides if the submissions are in parallel or sequential. Select **Yes**, if you want to run the flow instances in parallel. **Note:** It's recommended to run the threads in parallel for large volumes of data.
Maximum Parallel Threads	Maximum number of submissions executed in parallel.

Note: The **Maximum Repeat Counter** is the maximum number of instances that can be executed for a single submission of the task. It's the number of QuickPays processed in the payroll period and you intend to roll back using this flow. If the iteration runs into an error, this parameter prevents the process from getting into an infinite loop.

4. Click **Submit**.

When you submit a flow, you're taken to the Checklist page so that you can manage and monitor the tasks and other parameters within the flow.

Run the Archive EOY Payroll Results Report for Multiple PSUs

In this example you create a flow pattern and use the **Submit Another Task** task to automatically generate the Archive End-of-Year (EOY) report for multiple PSUs within your organization.

Before You Begin

Before you submit the task iterations to generate the report for the multiple PSUs, review and validate the year-end data and complete balance adjustments and balance feeds for year-end reporting.

Here are the key decisions for this example.

- The static flow parameters used in this example include Effective Date, Start Date, Tax Year Date, Repeat Counter.
- The dynamic flow parameters used in this example include Payroll Statutory Unit ID and the Repeat Flow.

The input parameters for the repeat submissions are obtained from the values returned by the task repeat formula. Perform these tasks to use Submit Another Task and generate the Archive End-of-Year (EOY) report for multiple PSUs.

1. Create a Task Repeat Formula

Use the text editor to create a fast formula and return the values required to run the Archive EOY Results for a PSU. This formula isn't specific to any legislative data group (LDG) and the formula type you use for this formula is Task Repeat.

Complete these steps to create a task repeat formula.

1. Navigate to My Client Groups < Show More < Payroll and use the **Fast Formulas** task.
2. On the Fast Formulas page, click **Create** to create a formula.
3. On the Create Fast Formula page, complete these fields.

Create Task Repeat Formula

Field	Value
Formula Name	Sample Formula
Formula Type	Task Repeat
Description	Archive EOY Results for each PSU
Effective Start Date	Enter a date, for example, January 01, 2011

4. Click **Continue**.
5. Enter this formula text details in the Formula Text Section.

```
FORMULA NAME: Sample Formula FORMULA TYPE: Task Repeat DESCRIPTION: Formula to iterate the
 EOY Archiver Formula Results: Iterates the EOY and generates the report /* Inputs */ INPUTS
 ARE REPEAT_COUNTER, BASE_TASK_NAME (text) REPEATFLAG = 'N' START_DATE = '2011-01-01'
 EFFECTIVE_DATE = '2012-01-01' TAX_YEAR_DATE = '2011-01-01' /* FORMULA BODY */ IF
 REPEAT_COUNTER= 1 THEN(PAYROLL_STATUTORY_UNIT = 300100001794785 REPEATFLAG = 'Y') IF
 REPEAT_COUNTER= 2 THEN(PAYROLL_STATUTORY_UNIT = 300100002950763 REPEATFLAG = 'Y') IF
 REPEAT_COUNTER= 3 THEN(PAYROLL_STATUTORY_UNIT = 300100013071724 REPEATFLAG = 'Y') IF
 REPEAT_COUNTER= 4 THEN(PAYROLL_STATUTORY_UNIT = 300100007796226 REPEATFLAG = 'N')
 /*Results*/ RETURN START_DATE, EFFECTIVE_DATE, TAX_YEAR_DATE, PAYROLL_STATUTORY_UNIT,
 REPEAT_COUNTER, REPEATFLAG /* End Formula Text */
```

6. Click **Compile**.
7. Click **Save**.

2. Create a Flow Pattern

Complete these steps to create a flow pattern.

1. Navigate to My Client Groups < Show More < Payroll and use the **Payroll Flow Patterns** task to create a flow pattern. You can also search for and select an existing flow pattern to copy.
2. Leave the Legislative Data Group field blank and click **Continue**.

3. On the Basic Information page, complete these basic flow information fields.

Create Flow Pattern

Field	Value
Flow Pattern Name	EOY Results Flow
Description	Enter a description for the flow.
LDG Required	No
Activities to Include	Select two options, Statutory and Calculate.

4. Select **Submit Another Task** to include it in the flow pattern.
5. On the Task Sequence page, reorder, add, or delete tasks as required.
6. Specify the order in which the tasks display in the checklist. You can specify a value for the sequence on the Edit Task Details Owners and Checklist page.
7. On the Parameters page, select **Create**.
8. On the Select and Add: Parameters page, add these parameters to complete the flow pattern.

1.

Name	Description	Task	Display
Task Name	Name of the task	Submit Another Task	Required
Task Repeat Formula	Name of the formula	Submit Another Task	Required
Maximum Repeat Counter	Maximum number of submissions	Submit Another Task	Required
Execute in Parallel	Decides if the submissions are in parallel or sequential	Submit Another Task	Yes
Maximum Parallel Threads	Maximum number of submissions executed in parallel	Submit Another Task	Yes

The flow parameters are used to submit and complete the tasks in the flow pattern, or as a basis for deriving values to submit the remaining tasks in the flow pattern.

2. After you have completed the requisite parameters, click **OK**.
3. On the Task Parameters page, review the parameters, and if necessary update the parameters.
4. Review the resulting checklist for the flow pattern before submitting the flow pattern.
5. Click **Submit**.

Submit the Flow

Complete these steps to submit the newly created flow pattern.

1. Select the **Submit a Flow** task from Quick Actions or under **Payroll** in **My Client Groups**.
2. Search for EOY Results Flow and click **Next**.
3. Enter these parameters.

Field	Value
Payroll Flow	Enter a payroll flow name.
Schedule	You can select 'As soon as possible' to execute the task immediately.
Task Name	Archive End-of-Year Payroll Results
Task Repeat Formula	Archive EOY Results Repeat Formula
Maximum Repeat Counter	Maximum number of iterations by the flow. In this example, it's the number of PSUs. **Note:** A repeat counter N indicates one parent and N-1 child submissions to iterate the task.
Execute in Parallel	Decides if the submissions are in parallel or sequential. Select **Yes**, if you want to run the flow instances in parallel. **Note:** For large volumes of data, it's recommended to run the threads in parallel.
Maximum Parallel Threads	Enter the number of threads executed in parallel.

Note: The Maximum Repeat Counter is maximum number of iterations that can be executed for a single submission of the task. It's the number of PSUs in the organization that you want to run the EOY Archiver. If the iteration runs into an error, this parameter prevents the process from getting into an infinite loop.

4. Click **Submit**.

When you submit a flow, you're taken to the Checklist page so that you can manage and monitor the tasks and other parameters within the flow.

3 Monitor Task Iterations and View Results

Monitor Task Iterations

When you submit a flow, you're taken to the Checklist page so that you can manage and monitor the tasks and other parameters within the flow.

On the Checklist page, search for and click on the Submit Another Task flow you just submitted. You're taken to the **Submit Another Task** page.

The page lists the multiple instances of the task iterations submitted by the Task Repeat Formula used in the flow. Use the Submit Another Task page to review and monitor the progress of the task iterations you have submitted and take corrective action as required.

- The Submit Another Task page displays each iteration task results as a separate row.
- To view further information for a specific task iteration, click on the task iteration name.
- Click **Refresh** to refresh the page and view the latest number of task iterations.
- Click **Actions > Roll Back All** to roll back all the task iterations.
- The **Parameters** section displays the parameters of the Submit Another Task flow.

View Task Iteration Results

After you subnit the flow, navigate to My Client Groups > Show More > Payroll and use the **Process Results Summary** task and search for the submitted Submit Another Task flow.

Use the Submit Another Task page to view the details of each task iteration of the single submission Submit Another Task flow.

- The **Submit Another Task** page displays each iteration task results as a separate row.
- Click on a task iteration name to navigate to the Process Results page for the specific iteration. You can then view the results details of each employee included in the task iteration.
- Use the tabs on the page to view specific info regarding the task iteration.
- Use the **Attachments** tab to access all attachments for the process such as reports.
- Use the **Logs** tab to view details of the processes and sub-processes included in the process. Use this data to troubleshoot processing errors.
- Use the **Parameters and Process History** tab to view the details of the parameters used to submit the process and view the process history. The process history details are used for audit purposes and to troubleshoot issues.
- Use the **Messages** tab to view messages related to the process or any of the sub processes.