

Workflows

Module Objectives

In this module, you will look at:

- What are Workflows
 - Creating a Workflow
 - Implementing a Workflow
-

Workflows

- **Workflows** are used to execute a series of *MapReduce* jobs
- A Workflow is given a sequence of jobs that follow each other, with an optional schedule to run the Workflow periodically
- **On successful execution** of a job, the control is transferred to the next job in sequence until the last job in the sequence is executed
- **On failure**, the execution is stopped at the failed job and no subsequent jobs in the sequence are executed

To process one or more MapReduce jobs in sequence, specify `withWorkflows()` in your application:

```
public ApplicationSpecification configure() {  
    return ApplicationSpecification.Builder.with()  
        ...  
        .withWorkflows()  
        .add(new PurchaseHistoryWorkflow())  
}
```

Implement the Workflow Interface

- Requires the `configure()` method
- From within `configure`, call the `addSchedule()` method to run a Workflow job periodically:

```
public static class PurchaseHistoryWorkflow implements Workflow {

    @Override
    public WorkflowSpecification configure() {
        return WorkflowSpecification.Builder.with()
            .setName("PurchaseHistoryWorkflow")
            .setDescription("PurchaseHistoryWorkflow description")
            .startWith(new PurchaseHistoryBuilder())
            .then(new PurchaseHistoryAggregateBuilder())
            .last(new PurchaseTrendBuilder())
            .addSchedule(new DefaultSchedule("FiveMinuteSchedule", "Run every 5 minutes",
                "0/5 * * * *", Schedule.Action.START))
            .build();
    }
}
```

Workflow with a single MapReduce Job

If there is only one MapReduce job to be run as a part of a WorkFlow, use the `onlyWith()` method after `setDescription()` when building the Workflow:

```
public static class PurchaseHistoryWorkflow implements Workflow {

    @Override
    public WorkflowSpecification configure() {
        return WorkflowSpecification.Builder.with() .setName("PurchaseHistoryWorkflow")
            .setDescription("PurchaseHistoryWorkflow description")
            .onlyWith(new PurchaseHistoryBuilder())
            .addSchedule(new DefaultSchedule("FiveMinuteSchedule", "Run every 5 minutes",
                "0/5 * * * *", Schedule.Action.START))
            .build();
    }
}
```

Module Summary

You should be able to:

- Define a Workflow
 - Create Workflows
 - Implement Workflows
-

Module Completed