**Amazon SWF----** Amazon SimpleWorkflow

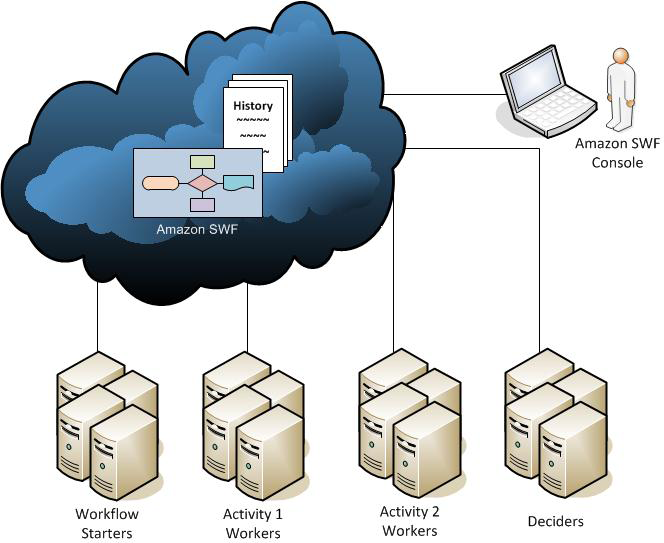
**What is a workflow**

A workflow is a sequence of multiple activities aimed at accomplishing a well-defined objective. For instance, booking an airline ticket as a workflow may encompass multiple activities, such as selection of itinerary, submission of personal details, payment validation and booking confirmation.

Except for the start and completion of a workflow, each step has a well-defined predecessor and successor. With that

* on successful completion of an activity the workflow can progress with its execution,
* when one of workflow’s activities fails it can be retried,
* and when it keeps failing repeatedly the workflow may regress to the previous step to gather alternative inputs or it may simply fail at that stage.

The Amazon Simple Workflow Service (Amazon SWF) makes it easy to build scalable distributed applications that coordinate work across distributed components. In this blog I will share a simple C# program to illustrate basic concepts*.*

 shows the Amazon SWF architecture, including Amazon SWF and its actors

## Workflow

The fundamental concept in Amazon SWF is the workflow. A workflow is a set of activities that carry out some objective, together with logic that coordinates the activities. For example, a workflow could receive a customer order and take whatever actions are necessary to fulfill it. Each workflow runs in an AWS resource called a domain, which controls the workflow's scope. An AWS account can have multiple domains, each of which can contain multiple workflows, but workflows in different domains cannot interact.

## Workflow Starters

A workflow starter is any application that can initiate workflow executions.

## Activity Workers

An activity worker is a process or thread that performs the activity tasks that are part of the workflow. (i.e.) Activity task is one step in the workflow. To use an activity task, you must register it using either the Amazon SWF console or the RegisterActivityType action.

Each activity worker polls Amazon SWF for new tasks that are appropriate for that activity worker to perform; certain tasks can be performed only by certain activity workers. After receiving a task, the activity worker processes the task to completion and then reports to Amazon SWF that the task was completed and provides the result. The activity worker then polls for a new task. The activity workers associated with a workflow execution continue in this way, processing tasks until the workflow execution itself is complete. Activity workers can run in AWS or in your datacenter behind the firewall.

## Deciders

A decider is an implementation of a workflow's coordination logic. Deciders control the flow of activity tasks in a workflow execution. Whenever a change occurs during a workflow execution, such as the completion of an activity task, Amazon SWF creates a decision task that contains the workflow history up to that point in time and assigns the task to a decider. When the decider receives the decision task from Amazon SWF, it analyzes the workflow execution history to determine the next appropriate steps in the workflow execution. The decider communicates these steps back to Amazon SWF using decisions. A decision is an Amazon SWF data type that can represent various next actions.

# Workflow Execution

1.       Write activity workers that implement the processing steps in your workflow.

2.       Write a decider to implement the coordination logic of your workflow.

3.       Register your activities and workflow with Amazon SWF. You can do this step programmatically or by using the AWS Management Console.

4.       Start your activity workers and decider. These actors can run on any computing device that can access an Amazon SWF endpoint. For example, you could use compute instances in the cloud, such as Amazon Elastic Compute Cloud (Amazon EC2); servers in your data center; Once started, the decider and activity workers should start polling Amazon SWF for tasks.

5.       Start one or more executions of your workflow. Executions can be initiated either programmatically or via the AWS Management Console. Each execution runs independently and you can provide each with its own set of input data. When an execution is started, Amazon SWF schedules the initial decision task. In response, your decider begins generating decisions which initiate activity tasks. Execution continues until your decider makes a decision to close the execution.

6.       View workflow executions using the AWS Management Console. You can filter and view complete details of running as well as completed executions. For example, you can select an open execution to see which tasks have completed and what their results were.

**How does Amazon SWF help you accomplish this**

Amazon SimpleWorkflow service defines an interface for workflow orchestration and provides state persistence for workflow executions.

Amazon SWF applications involve communication between the following entities:

* The Amazon Simple Workflow Service - providing centralized orchestration and workflow state persistence,
* Workflow Executors - some entity starting workflow executions, typically through an action taken by a user or from a cronjob.
* Deciders - a program codifying the business logic, i.e. a set of instructions and decisions. Deciders take decisions based on initial set of conditions and outcomes from activities.
* Activity Workers - their objective is very straightforward: to take inputs, execute the tasks and return a result to the Service.