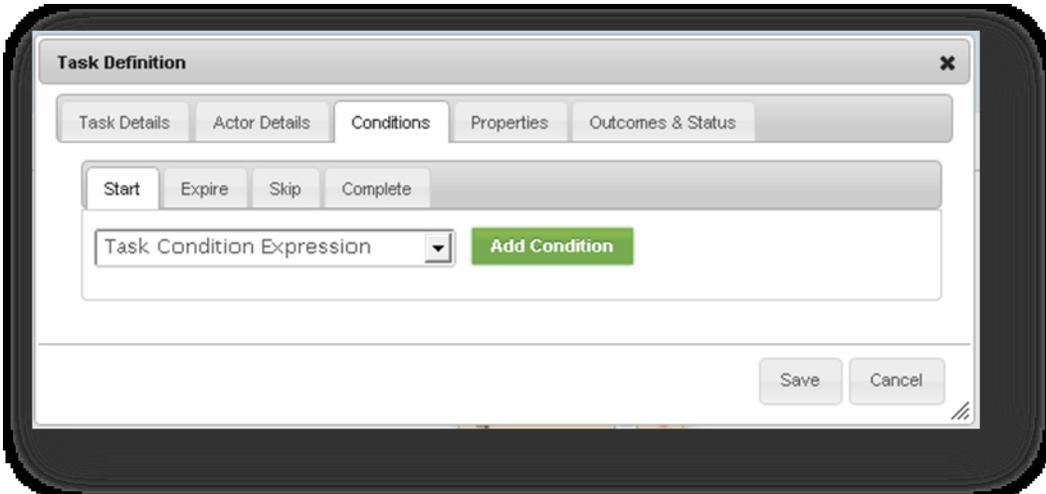


# Cello How-To Guide

Configure / Manage Workflow

## Contents

- 1 Introduction to Cello Workflow ..... 4
  - 1.1 Sample Use Cases..... 4
  - 1.2 CelloSaaS Workflow ..... 6
  - 1.3 Steps in a Workflow ..... 6
  - 1.4 Designer Tools..... 7
  - 1.5 Execution of Tasks..... 8
  - 1.6 Out-of-Box Activities ..... 8
  - 1.7 Activity Definitions ..... 9
  - 1.8 Activity Inputs ..... 9
  - 1.9 Default Activities..... 10
    - 1.9.1 Email Notification Activity..... 10
    - 1.9.2 Approval Activity ..... 10
    - 1.9.3 Service Call Activity ..... 11
    - 1.9.4 Activity Output ..... 12
    - 1.9.5 Step Router ..... 12
  - 1.10 Workflow Instance, Step Instance and Step routing ..... 13
  - 1.11 Managing Workflows ..... 14
- 2 How to design a Workflow..... 15
  - 2.1 Editing a workflow ..... 17
  - 2.2 Managing Workflow Designs ..... 17
  - 2.3 Manual and Auto Task Containers ..... 18
  - 2.4 Task Definition Properties..... 19
  - 2.5 Task Definition Properties..... 20



..... 21

2.6 Sample Business Rule Condition ..... 21

2.7 Task Definition Properties..... 22

3 Benefits of CelloSaaS Workflow..... 25

4 Contact Information..... 26

## 1 Introduction to Cello Workflow

Workflow is a term used to describe the tasks, procedural steps, organizations or people involved, required input and output information, and tools needed for each step in a business process. A workflow approach to analyzing and managing a business process can be combined with an object oriented programming approach, which tends to focus on documents and data.

In general, workflow management focuses on processes rather than Objective.

For example, an insurance company could use the CelloSaaS workflow Engine to ensure that a claim was handled consistently from initial call to final settlement. The CelloSaaS workflow Engine would ensure that each person handling the claim used the correct online form and successfully completed their step before allowing the process to proceed to the next person and procedural step.

### 1.1 Sample Use Cases

Cello Workflow Management system allows the individual Tenant to design different workflows for different types of jobs or processes.

#### Use Case 1:

In a manufacturing setting, a design document might be automatically routed from designer to a technical director to the production engineer.

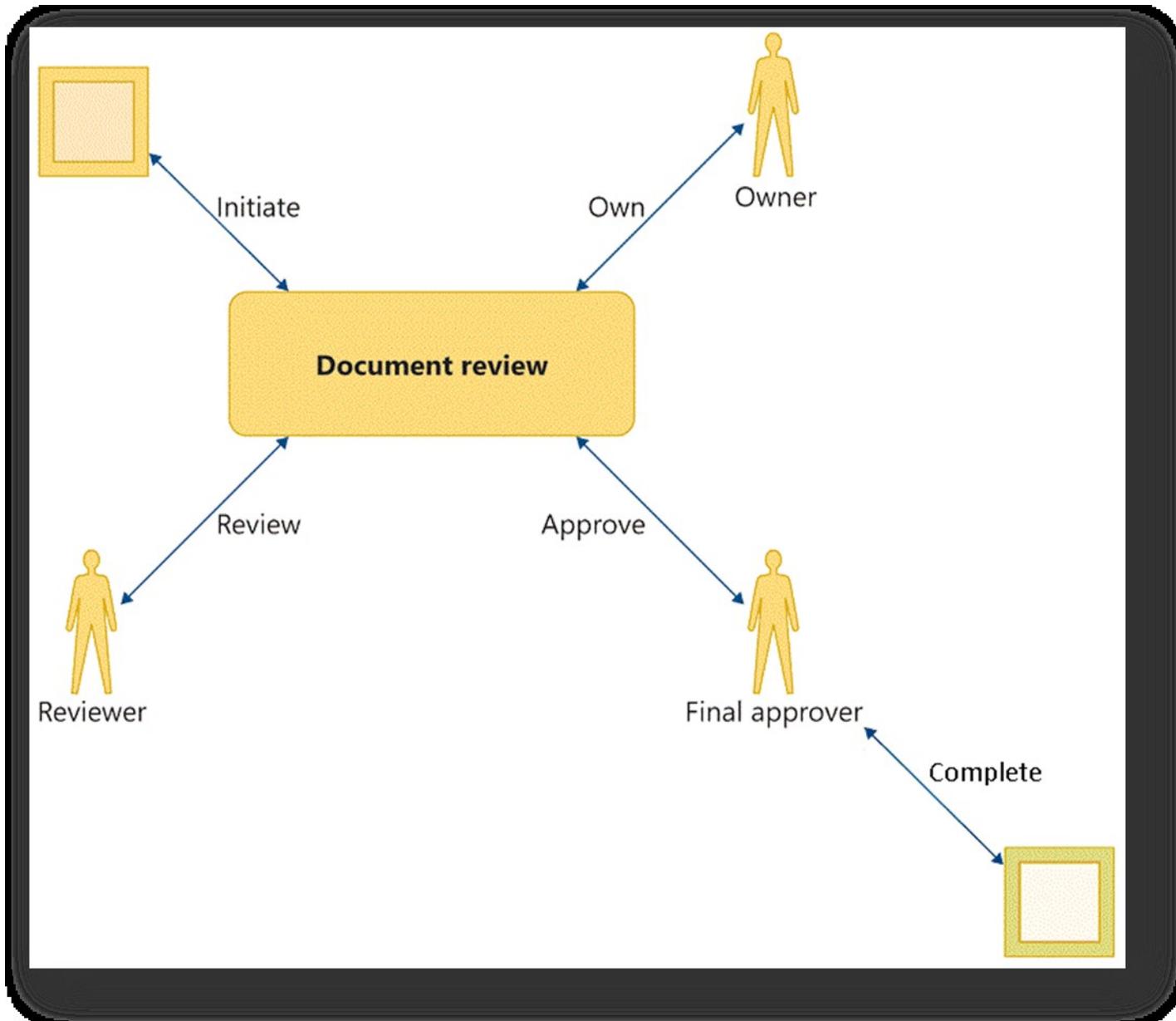
At each stage in the workflow, one individual or group is responsible for a specific task. Once the task is complete, the workflow ensures that the individuals responsible for the next task are notified and receive the data they need to execute their stage of the process. Workflow management systems also automate redundant tasks and ensure that uncompleted tasks are followed up.

#### Use Case 2:

Suppose that a document review process specifies that a user named Joe writes the document and then Fred reviews it. Here, the work units are first writing and second reviewing the document, and the organization is that one task must follow the other. Code that makes successive calls to two subroutines is a valid example of the concept. The interest lies rather in the forms that this organization takes.

A *document review* process takes as an input parameter a set of [reviewer, role] pairs that describe which humans are involved in the workflow in which roles. Possible values for the role are required, optional, final approver, and owner. The review process then proceeds until all reviewers have performed their assigned roles and notify the owner of the outcome.

Here, the work items are the document reviews organized by the review process. There are three interesting characteristics to call out, namely, multiple points of interaction, human and automated activity, and the need to handle dynamic change.



**Sample Document Review Workflow**

## Pointers

[1] Each activity has a set of outcomes. These activities and their outcomes are mapped in the Activities Table in CelloSaaS.

[2] Cello allows Tenant based custom workflow design for a single workflow and also maintain different workflows for different tasks. Also workflow can be grouped by custom categories.

[3] This requirement is captured via the task actors or dynamic task actors. There are two options, namely usernames based user assignment or role based user assignment.

[4] These are the tasks which are associated with the activities like manual / automated task with their corresponding activities.

[5] Each task can take inputs and is defined by the developer and is made available via Inputs to activities in a task.

- [6] Cello identifies the users by either usernames or the users that belong to a role, identified by role names.
- [7] These process can be globally defined and tenant-wise customizable. If not customized, the workflow engine defaults to the global process [workflow].
- [8] There are routers that decide on what step to proceed next based on the outcomes of the previous conditions. Cello has different types of Routers to accomplish this feature.
- [9] This is achieved via the Manual Task. This will require a manual intervention to process the activity and to move it to next step.
- [10] This is achieved via the Automatic / Automated Task. No manual intervention is required. This can be an auto approval based on the outcome of the previous task.

## 1.2 CelloSaaS Workflow

- 1 CelloSaaS provides design time and runtime support for workflow management.
- 2 A workflow can be invoked based on the following
  - a Occurrence of an event
  - b Invocation in the code as part of a business process
- 3 Workflows should be uniquely identified
- 4 Workflows will have an input on which the workflow acts
- 5 Any user who has privilege to access Workflow Module can design a workflow with the steps registered
- 6 The workflow can be versioned for each tenant
- 7 The workflow version can be published and when reopened for editing it gets to the draft mode

## 1.3 Steps in a Workflow

- 1 Workflow is an orchestration of steps crafted to portray the business process
- 2 Every workflow begins with a **start** step and ends with a **stop** step
- 3 A step can either be manual or automatic
- 4 A step must be mapped to an activity and can further have provisions to associate task or code conditions for Skip, Start, Expiry and Complete conditions. All the conditions can either be
  - o Task outcome
  - o Execution based conditions
  - o Code condition
  - o Business Rule conditions.
- 5 Every step can have one or more outcomes returned by the corresponding activities.
- 6 The step status can
  - o Routed : Step is routed and is awaiting processing by the next step
  - o Skipped: This step is skipped from execution based on the step's associated condition
  - o Started: This step has started processing
  - o Executed: This step has completed execution
  - o Errored: Some exception has occurred impeding the execution of this step
  - o Completed: This step has run to completion
  - o Expired: This step failed to execute / complete within the specified expiry condition and is considered to be expired. This step can be re-started or revived manually.

- 7 A manual step can also contain definition for who all should act on that step. This can either be static roles or dynamic users.

## 1.4 Designer Tools

Tool	Description
	Start Step: Every workflow begins with a Start Step and this step is mandatory
	Manual Task represents the task that requires user intervention. This task also obtains the users that will be required to act on the task.
	Auto-Task represents the task that can execute based on the previous task outcome or conditions without requiring user intervention.
	Router is used to route the flow to tasks based on some conditions.
	If-Else-Router is a specialized form of Router that has if-else branching construct for the conditions. If the condition is true then task which is connected with If node will be executed and vice-versa..



End Step: Every workflow ends with this End Step. This step is mandatory as in the case with the Start Task.

## 1.5 Execution of Tasks

CelloSaaS supports the following types of tasks in a workflow.

- a Manual Task - Refers to the task that requires a manual / user involvement to continue processing.
- b Auto Task – Refers to a task which doesn't require manual / user involvement to process.

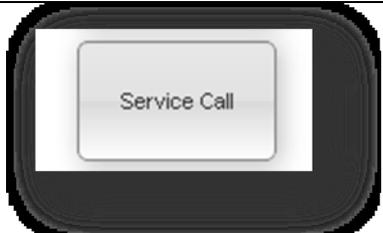
## 1.6 Out-of-Box Activities

A container(Manual/Automated) holds an activity defined in a order in which the user should interact with. The Activities toolbar lists all the activities that are registered with Workflow. A task will refer to an activity which could be any kind of action.

**For Example:**

Sending email, Update the database, Processing document or image etc.

- All possible activities/jobs need to be thought through and developed and registered
- An activity can be used in any number of steps within a workflow
- The stored procedures that can be utilized for the activities should also be registered

Activity Tools *	Description
	This Activity uses the CelloSaaS notification manger to send an email Notification based on the configured properties in this activity
	This activity will make a service call based on the registered web service URL with the inputs and format the outputs for further use by workflow engine.



Performs the approval or reject action on the input based on the task conditions

**Note:** Custom activities that would be developed and registered with CelloSaaS workflow will also be listed. Refer [here](#) how to write Custom Activities with CelloSaaS Workflow.

## 1.7 Activity Definitions

The following are the details for the default activities that are supplied with CelloSaaS out of the box.

A screenshot of a web application interface for workflow management. The interface has a dark grey header with three tabs: "Workflow Details", "Design", and "Activities". The "Activities" tab is selected. Below the tabs, there are three sections, each with a blue header and a white background. The first section is titled "Email Notification" and contains the following text: "Description: Sends Notification", "Type Name: CelloSaaS.WorkFlow.Activities.NotificationActivityDefinition", "Assembly Name: CelloSaaS.WorkFlow", "Outcomes: Complete", and "Properties: Notification Name , Subject , To , To xPath , To Property , Cc , Cc xPath , Cc Property". The second section is titled "Service Call" and contains: "Description: Calls a service through web service", "Type Name: CelloSaaS.WorkFlow.Activities.ServiceCallActivityDefinition", "Assembly Name: CelloSaaS.WorkFlow", "Outcomes: Complete", and "Properties: Service Name , Service Input Transform , Workflow Input Transform". The third section is titled "Approval" and contains: "Description: Approval Activity", "Type Name: CelloSaaS.WorkFlow.Activities.ApprovalActivityDefinition", "Assembly Name: CelloSaaS.WorkFlow", "Outcomes: Approve,Reject", and "Properties: -".

## 1.8 Activity Inputs

The inputs that may be passed to an activity can be either objects or XML data. In case of an In-Proc mode, the data can be available in object format and in the case with Web Services, the inputs can be in XML format and CelloSaaS supports both these modes.

## 1.9 Default Activities

The following are the default activities shipped out of the box with CelloSaaS

### 1.9.1 Email Notification Activity

- a. This activity can be used to trigger an email notification.
- b. The following are the properties
  - i. **Notification Name** : Specify the notification name from notification management
  - ii. **Subject** : Specify the notification subject name
  - iii. **To, CC** : The inputs will typically be a comma delimited string or string array or a list of email ids. Sample: [user1@example.com](mailto:user1@example.com), [user2@example.com](mailto:user2@example.com)
  - iv. **ToXPath, CCXPath**: Typical inputs will be a XPath query to obtain the To, CC email ids from the Workflow input in case of the input being a WorkflowXmlInput.  
Sample: `//userDetails /MembershipDetails/ EmailId`
  - v. **To Property, CC Property**: In this property, we can provide the property of the application entity that can provide the values for the To / CC property.  
Sample: `userDetails.MembershipDetails.EmailId`

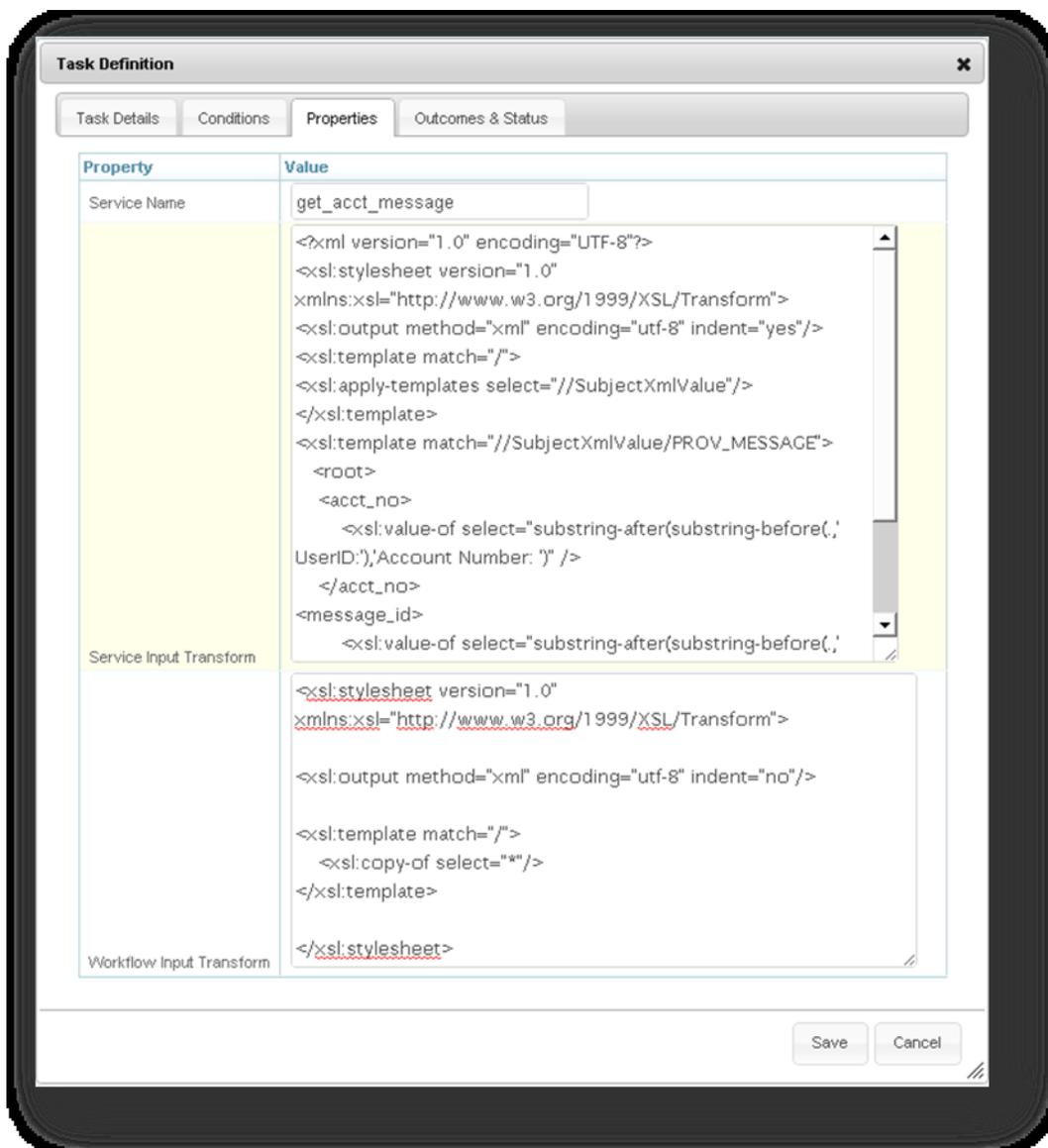
Property	Value
Notification Name	Approval Mail
Subject	Document is approved after Review
To	userone@example.com
To XPath	//userDetails /MembershipDetails/ EmailId
To Property	userDetails.MembershipDetails.EmailId
Cc	
Cc XPath	
Cc Property	

### 1.9.2 Approval Activity

- c. This activity can be used to either approve or reject a task.
- d. There are no properties involved in this activity as this activity can either approve or reject

## 1.9.3 Service Call Activity

- e. A web service can be invoked by providing the fully qualified service path in the Service Name property.
- f. The Service Input Transform is used to input the XSLT that can be used to transform the workflow input to service call input.
- g. The Workflow input transform is used to input the XSLT that can be used to transform the service call output to workflow input.



## 1.9.4 Activity Output

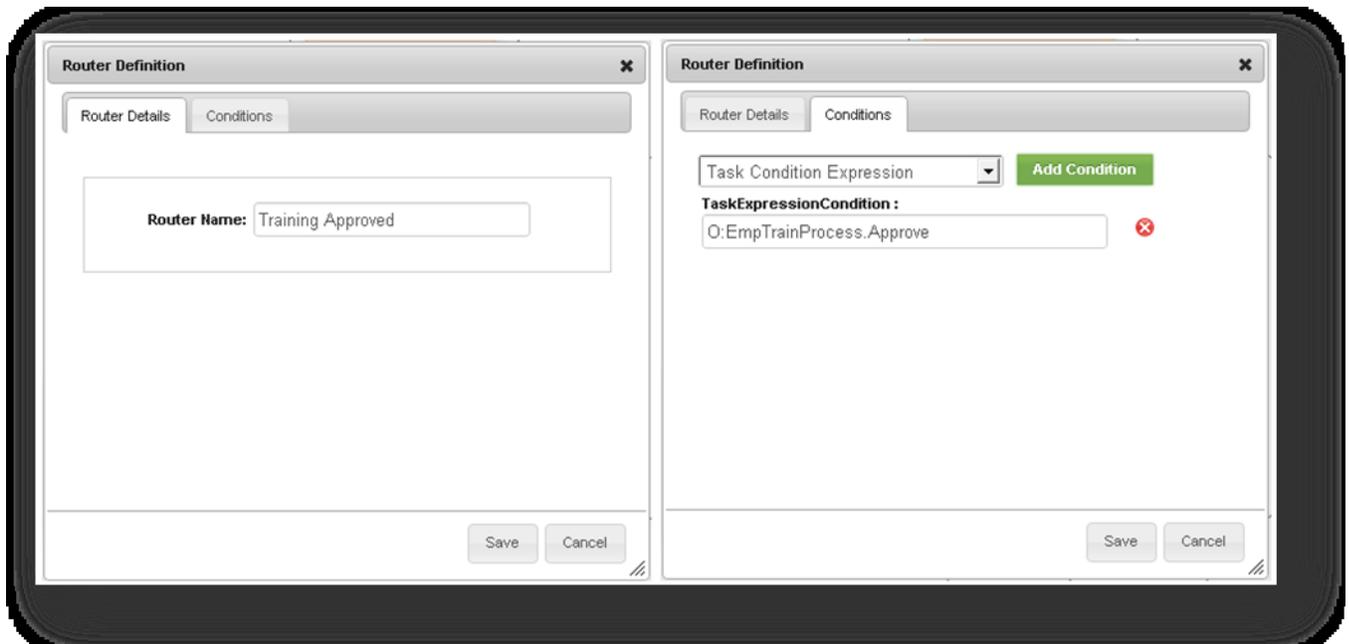
The following are the possible outcomes for the default activities from CelloSaaS.

Activity Name	Possible Outcomes
Email Notification	Complete
Service Call	Complete
Approval	Approve, Reject

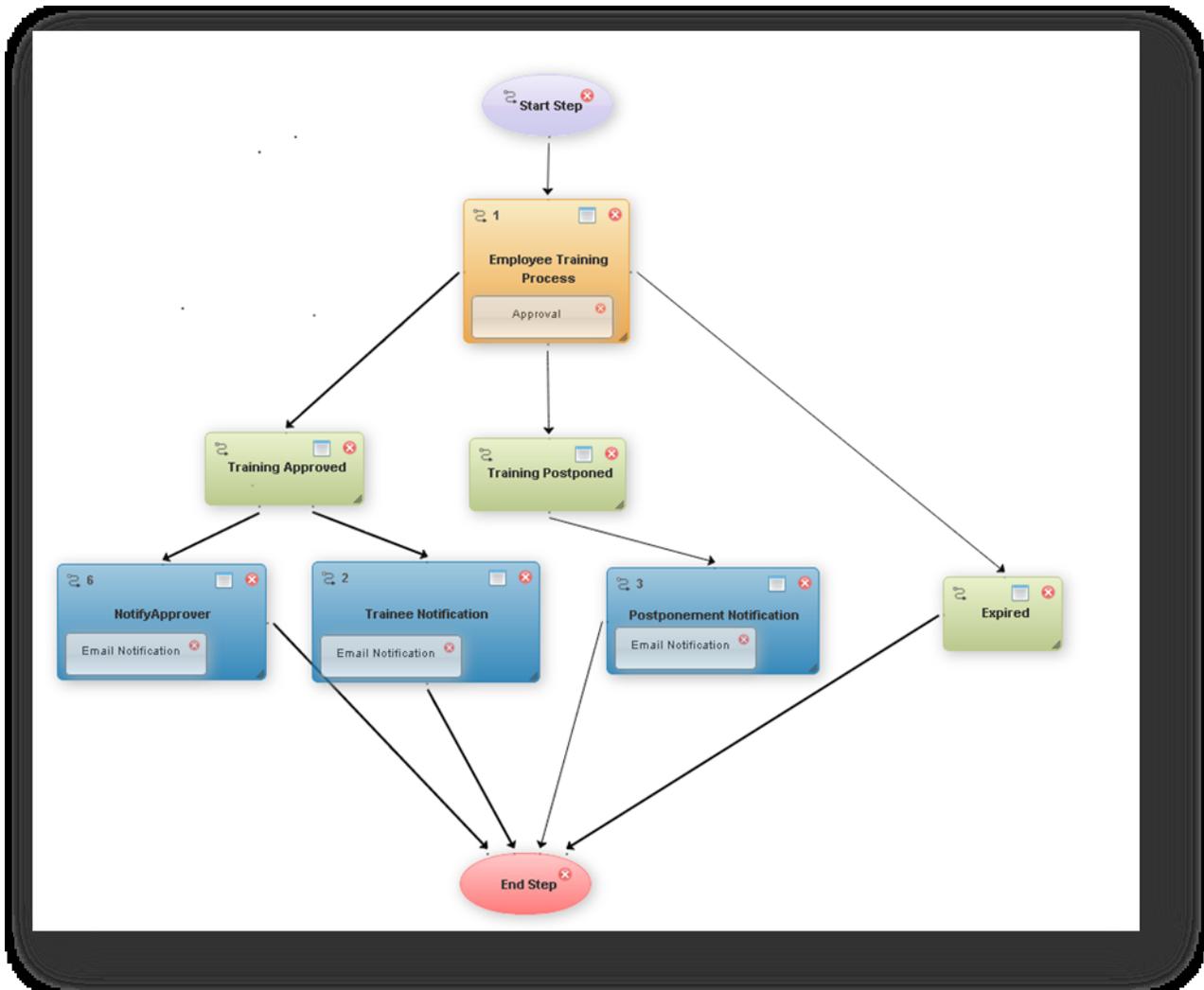
## 1.9.5 Step Router

- The step router is responsible for routing the flow between the steps in workflow
- This routing is based on task conditions which are again based on the previous step outcomes and execution status
- Step router can have multiple inputs and output flows.
- The router can have decisions which is a combination of outcomes from two tasks
- Parallel routing is possible
- Re-routing to a finished task is also possible

A sample router with the corresponding code expression condition is given below



The following sample training workflow shows the parallel routing in a Router



## 1.10 Workflow Instance, Step Instance and Step routing

- When the Workflow is invoked, an instance is created with a unique identification
- Each workflow status is tracked through various statuses that it may pass through as time progresses.
- Once the workflow instance starts it begins with the start step, once the start step is executed, it routes it to the router.
- The router then evaluates the next set of steps that need to be routed to.
- Once the steps are routed, the task instance gets to the status “Routed” and the workflow instance checks if the step can be skipped
  - If it can be skipped, the task instance moves to a status called Skipped and the corresponding routers linked to the task gets evaluated and then routed to.
  - If skip condition fails then the start task for this step will be called.
- If the step can be started, the task instance moves to a status called “started” and calls the execute action which internally executes the activity mapped to it.
- The execute activity returns back the outcomes and the status is updated to “executed”

- Once the task is executed, the step is checked if it can be completed. If so, it completes the task and marks the task instance status as “Completed”.
- If any exception occurs while executing the task then the task status is set as “Errored”.
- When the end step is completed then the workflow instance gets completed and its status is marked as “Completed”.
- If task complete condition fails then expired conditions will be checked. If expire condition is satisfied then the task status moves to “Expired”.
- Whenever there is any change in the status or outcome of the task status then the corresponding routers linked to the task gets evaluated and then routed to.

## 1.11 Managing Workflows

Navigate to **Workflow** → **Workflow**

CelloSaaS admin@company.com | Change Tenant

My Account Subscription Tenant Access Control Configuration Business Events Notifications Workflow Reporting Monitor

Manage Workflow

Search [GO]

Workflow List

Workflow Details Design Activities

Workflow Name: UserActivation

Description: UserActivation

Workflow Input Finder Type: [ ]

Category: User

XML Transform: 1

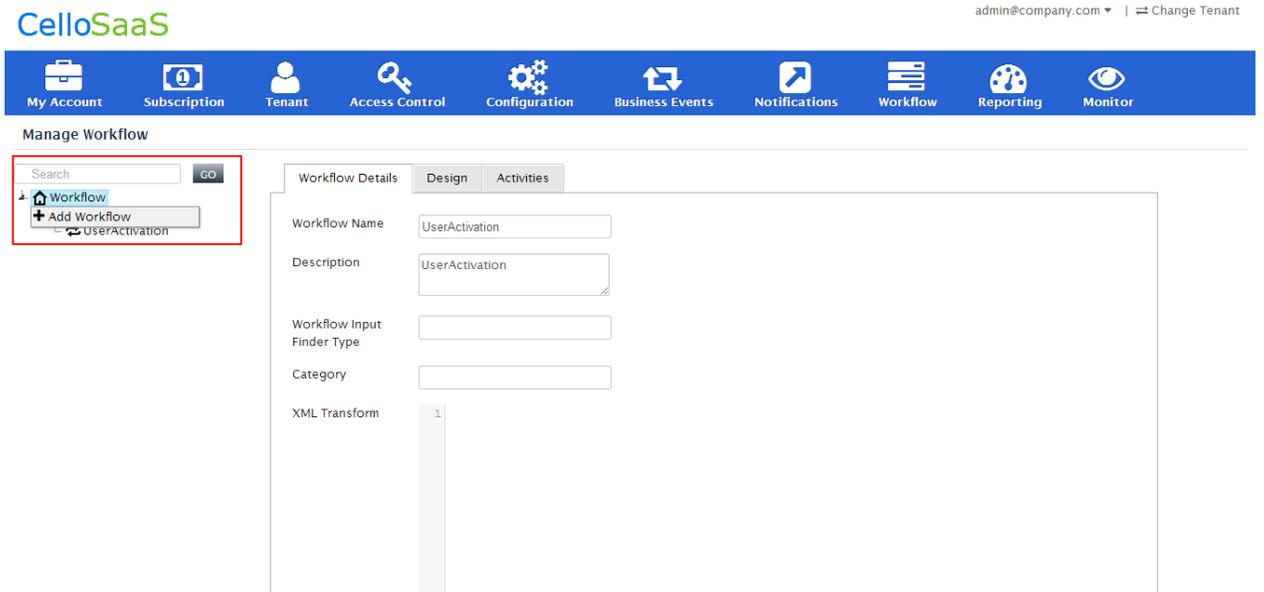
Workflow Home Page

Save

v 4.3.1.0 Copyright © 2013 by techcello.com All Rights Reserved.

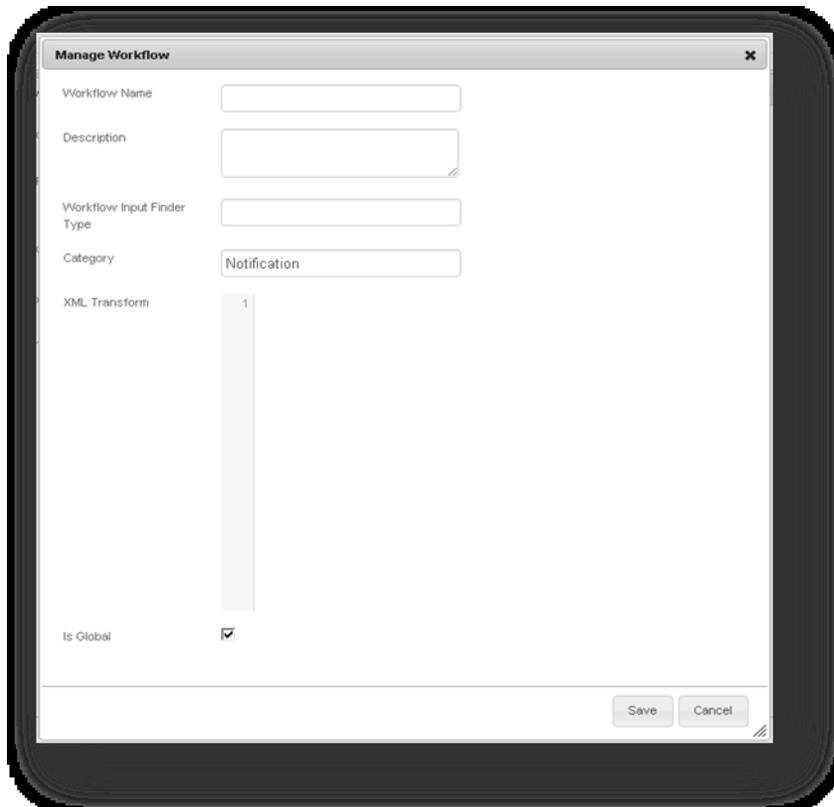
## 2 How to design a Workflow

Highlight the workflow home icon and then right click to get the context menu. The context menu lists the possible actions such as, “Add Workflow”. Click on the “Add Workflow” context menu item to add a new workflow.



**Note:** Workflows created from the above UI are created on behalf of the logged in tenant only.

The following screenshot illustrates the various inputs that are required to create a new workflow.



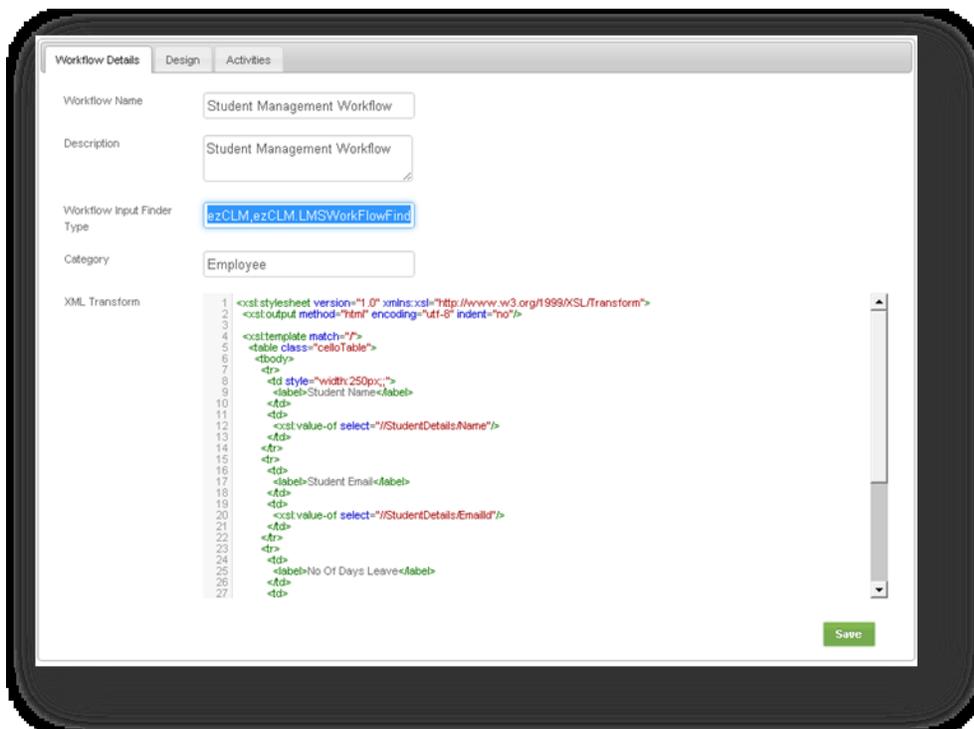
**Adding a new workflow**

Input	Description
Name	Workflow name [Unique per tenant]
Description	Short description about the workflow
Workflow Input finder Type	The Input finder class that will get the workflow input when the workflow is to resume with the data from the database.  (Fully Qualified Name Ex: <i>UserDetails.Contact</i> )
Category	Logical grouping of the workflow
XML Transform	The XSLT that is required to transform the XML input to the workflow input
Is Global	To indicate that this workflow is visible across all the tenants.

## 2.1 Editing a workflow

User can edit the workflow upon click the edit icon, the workflow details with workflow name, description, and active version are displayed. It has three different designs like Default Workflow Design, Published Workflow Design, and Un-Published Workflow Design.

- The Default Workflow Design contains version, published status, date created and updated. The Product Admin can only have the permissions to edit the default workflow design.
- The Published Workflow Design contains the list of published workflows. Through this the user can view the workflow design. Using copy option the tenant can copy the workflow and change it for own. And only one published workflow will be in Active status.
- The Un-Published Workflow Design lists out the workflows which are unpublished.



## 2.2 Managing Workflow Designs

In this page, the various orchestrated workflows are listed,

1. Default Workflow Design
  - a. This is the workflow design that is default and the tenant administrators can override this default design and create their own to suit their requirements. If the overridden one is not published, the default workflow is taken for execution.
2. Published Workflow Design

- a. This is the list of designs that have been published by the current tenant admin along with the version numbers.
3. Un-Published workflow design
  - a. This contains the designs that are yet to be published.

The screenshot shows a web interface for managing a workflow. At the top, there are tabs for 'Workflow Details', 'Design', and 'Activities'. The main heading is 'Manage 'Employee Training Workflow'' with a 'Create New' button and a 'Filter Id' dropdown menu.

Below the heading, there is a summary box with the following information:

- Workflow Name:** Employee Training Workflow
- Description:** Employee Training Workflow
- Active Version:** 14

The 'Default Workflow Design' section contains a table with the following data:

Version	Publish Status	Created On	Updated On	Edit	Copy
1	Published	1/29/2013 9:46:28 AM	1/29/2013 11:32:36 AM		

The 'Published Workflow Designs' section contains a table with the following data:

Filter Id	Version	Created On	Updated On	Status	View	Copy
---	14	2/1/2013 10:59:16 AM	2/1/2013 12:01:50 PM	Active	View	
---	13	2/1/2013 10:50:33 AM	2/1/2013 10:51:25 AM	---	View	
---	12	2/1/2013 10:42:56 AM	2/1/2013 10:43:35 AM	---	View	
---	11	1/31/2013 4:17:23 PM	2/1/2013 10:38:39 AM	---	View	
---	10	1/30/2013 3:31:25 PM	1/30/2013 3:31:27 PM	---	View	
---	9	1/30/2013 2:34:46 PM	1/30/2013 2:35:07 PM	---	View	
---	8	1/30/2013 2:31:51 PM	1/30/2013 2:32:31 PM	---	View	
---	7	1/30/2013 12:10:20 PM	1/30/2013 2:25:00 PM	---	View	
---	6	1/29/2013 4:52:18 PM	1/30/2013 12:06:07 PM	---	View	
---	5	1/29/2013 3:37:33 PM	1/29/2013 3:38:27 PM	---	View	
---	4	1/29/2013 3:32:24 PM	1/29/2013 3:34:04 PM	---	View	
---	3	1/29/2013 1:49:48 PM	1/29/2013 1:55:38 PM	---	View	
---	2	1/29/2013 11:32:41 AM	1/29/2013 12:15:46 PM	---	View	

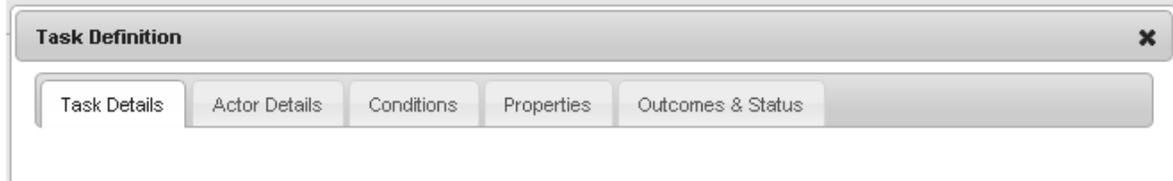
The 'Un-Published Workflow Design' section contains a table with the following data:

Filter Id	Version	Created On	Updated On	Action	Edit	Delete
---	15	2/1/2013 3:37:42 PM	--	Publish		

## 2.3 Manual and Auto Task Containers

The difference between a manual and an automatic task definition is that the manual task definition involves the actor detail capture to perform the manual task which is not the case with the automatic task.

## Manual Task Definition

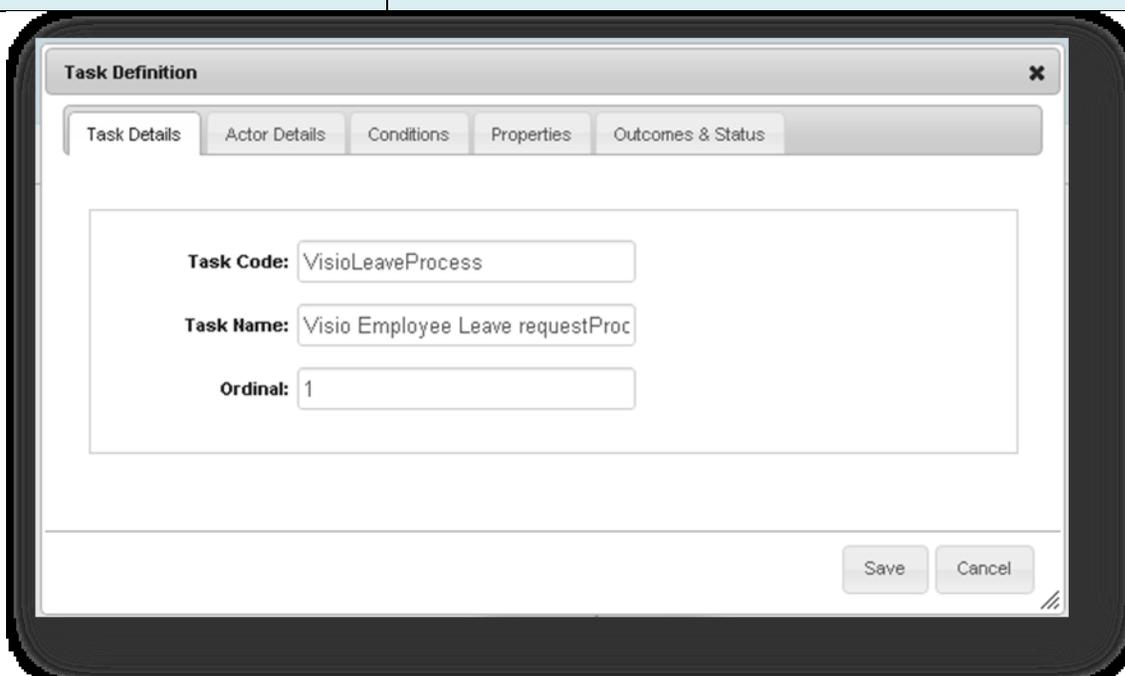


## Auto Task Definition



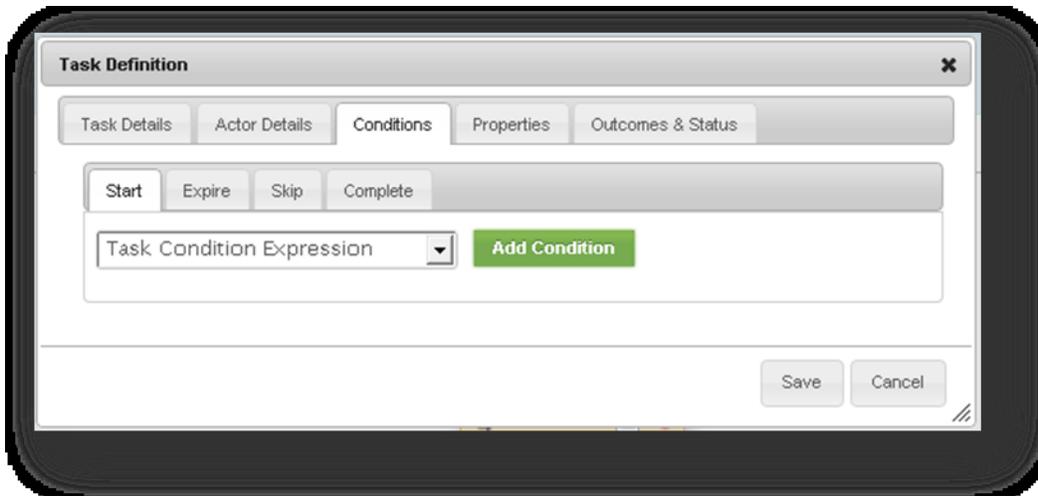
## 2.4 Task Definition Properties

Task	Description
<i>Task Code</i>	A unique task code in the workflow.
<i>Task Name</i>	Name of the Task
<i>Ordinal</i>	Determines the ordinal value of the Task. Default ordinal value will set while drag and drop. The user also can edit and change the ordinal values as like.



## 2.5 Task Definition Properties

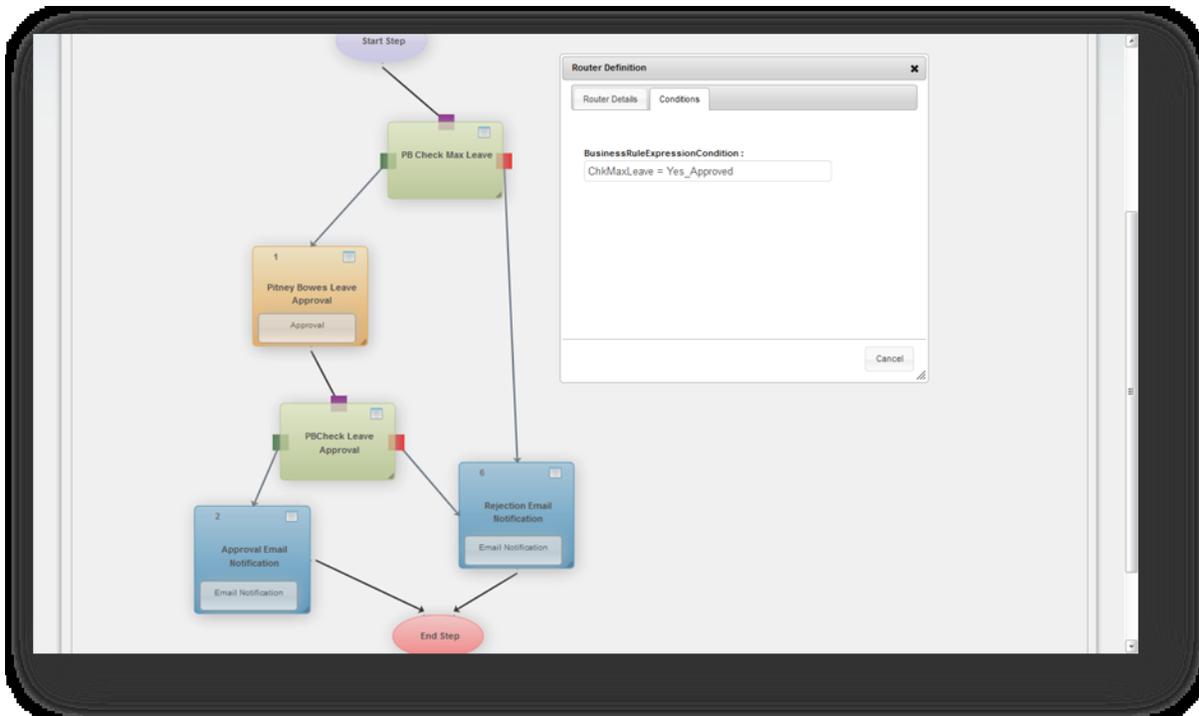
Task	Description
<p><i>Conditions</i></p>	<p>It has the conditions to execute on different process of the workflow execution like Start, Expire, Skip, and Complete. Each has the conditions for own</p> <ul style="list-style-type: none"> <li>○ The <b>Start conditions</b> will check before the task started. If the condition is true the task will get start.</li> <li>○ The <b>Expire conditions</b> will check before the task expired</li> <li>○ The <b>Skip condition</b> will check the condition while skips the task.</li> <li>○ The <b>Complete condition</b> will check the condition while the task is complete.</li> </ul>
<p><i>Conditions Type</i></p>	<p><b>Task Condition:</b></p> <ul style="list-style-type: none"> <li>○ It has the condition while the task is process. It has two types as outcomes and executions status.</li> <li>○ The Outcome based condition will be like "<i>O:[TaskName].[Outcome]</i>"                      O → outcome                      [TaskName] → Task name                      [Outcome] → Task possible outcomes from Execute and Complete methods.</li> <li>○ The Execution Status based condition will be like "<i>E:[TaskName].[ExecutionStatus]</i>" .                       E → Execution Status                       [TaskName] → Task name                       [ExecutionStatus] → Execution status.                       Possible execution status → Started, Skipped, Routed, Errored, Executed, Expired, Completed.</li> </ul> <p><b>Code Condition:</b></p> <ul style="list-style-type: none"> <li>○ The user can have their own conditions like "<i>Systems.DateTime.Now &gt; Convert.ToDateTime(dateTime)</i>" which is plain C# code.</li> <li>○ Entity also can be used in this condition.</li> </ul> <p><b>Stored Procedure Condition:</b></p> <ul style="list-style-type: none"> <li>○ We are not supporting this feature in this version.</li> </ul>



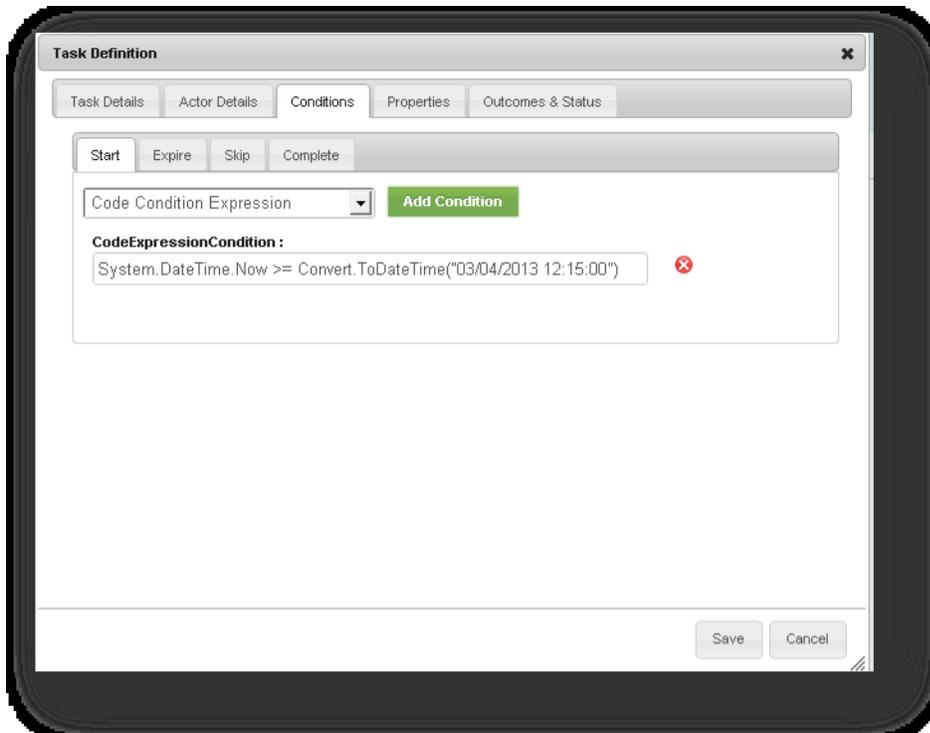
## 2.6 Sample Business Rule Condition

Business Rules Engine is one of the modules of CelloSaaS, and Business Rules created by the Cello BRE can be used as a condition for evaluation.

1. Mapping the Rule to the Workflow router condition

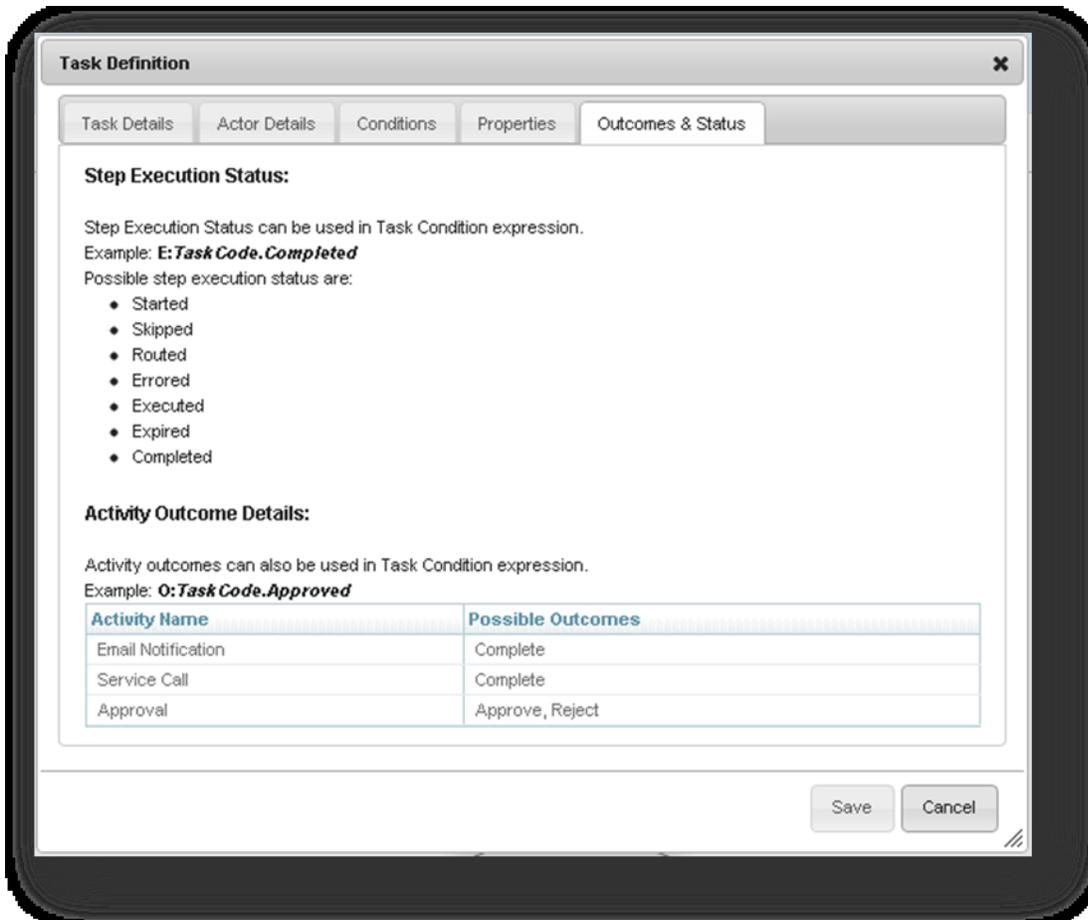


Sample Code Expression Condition



## 2.7 Task Definition Properties

Task	Description
Outcomes and Status	<i>The Outcomes and Status contains the possible outcomes list and possible execution status list. This outcomes and Executions status list will define while the activity creation</i>



Task	Description
<p><i>Actor Details</i></p> <p><i>(only for Manual Task)</i></p>	<p>It has <i>Roles</i>, <i>Role Names</i>, <i>User Names</i> and <i>Url</i> attributes. Roles textbox we will give the role identifiers with comma separated. Role Names textbox we will give the role names. User Names textbox we will give the user names of the tenant. The <i>Url</i> is used to execute the Task with use of given <i>Url</i> value.</p>

**Task Definition** [X]

Task Details | Actor Details | Conditions | Properties | Outcomes & Status

**Roles:**   
*(Enter role ids seperated by ',')*

**Role Names:**   
*(Enter role names seperated by ',')*

**User Names:**   
*(Enter user names seperated by ',')*

**Uri:**

Save Cancel

## 3 Benefits of CelloSaaS Workflow

- It's Web based
- Drag and Drop based Workflow Designer
- It's Multi Tenant
- Customize and Configure the Workflows using web based Designer
- No IT Skills required
- Support for Versioning
- Support for Activity/Workflow Monitoring
- Support for Manual & Automated Execution of Activities
- Support for Business Rules
- Service Orientation
- Support for Tenant Hierarchy
- Support for Design & Runtime
- Integration with Cello Business Events
- Integration with Cello Security System

## 4 Contact Information

Any problem using this guide (or) using Cello Framework. Please feel free to contact us, we will be happy to assist you in getting started with Cello.

**Email:** [support@techcello.com](mailto:support@techcello.com)

**Phone:** +1(609)503-7163

**Skype:** techcello