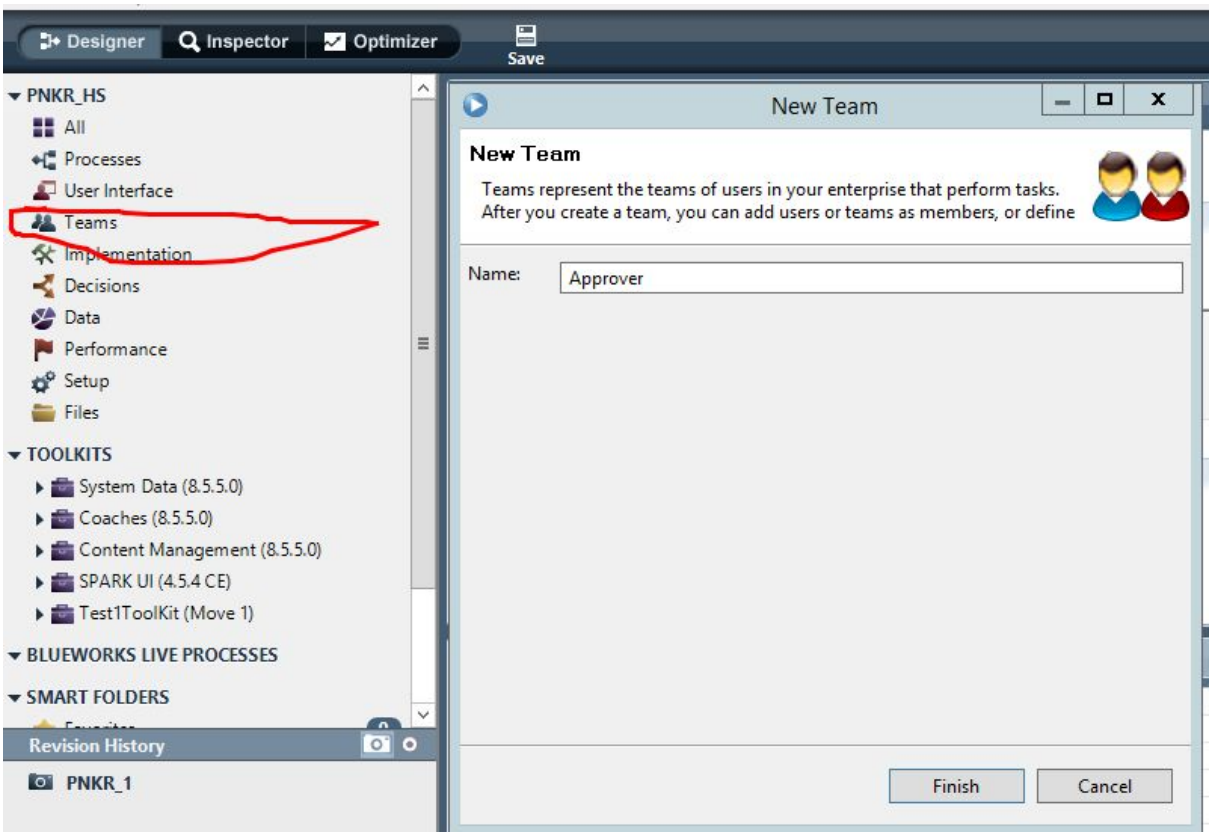
Team Filter and Team Retrieval Service

Description:Team retrieval and team filter services allow you to programmatically derive users and groups to use at runtime.Team retrieval service allows you to dynamically create a Team and Team Filter Service allows you to filer the Team based on the requirement.\

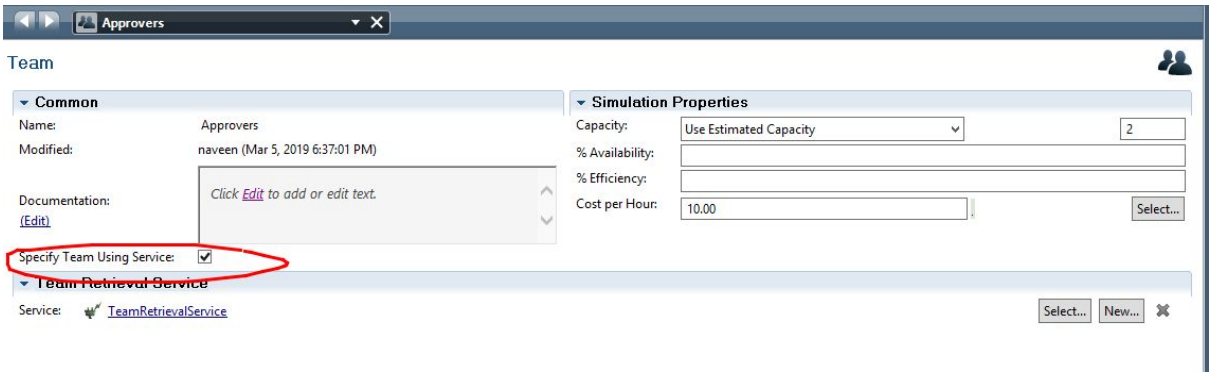
Team Retrieval Service:A team retrieval service is an implementation of an integration service based on the predefined Team Retrieval Service Template that is included in IBM BPM V8.5. The template mandates that the service returns a team parameter of the predefined BO (business object) Team type and takes a name (String) parameter as input. You cannot change the output definition for a team retrieval service. However, you can change the input definition by adding custom parameters to it.

Creating Team Retrieval Service:

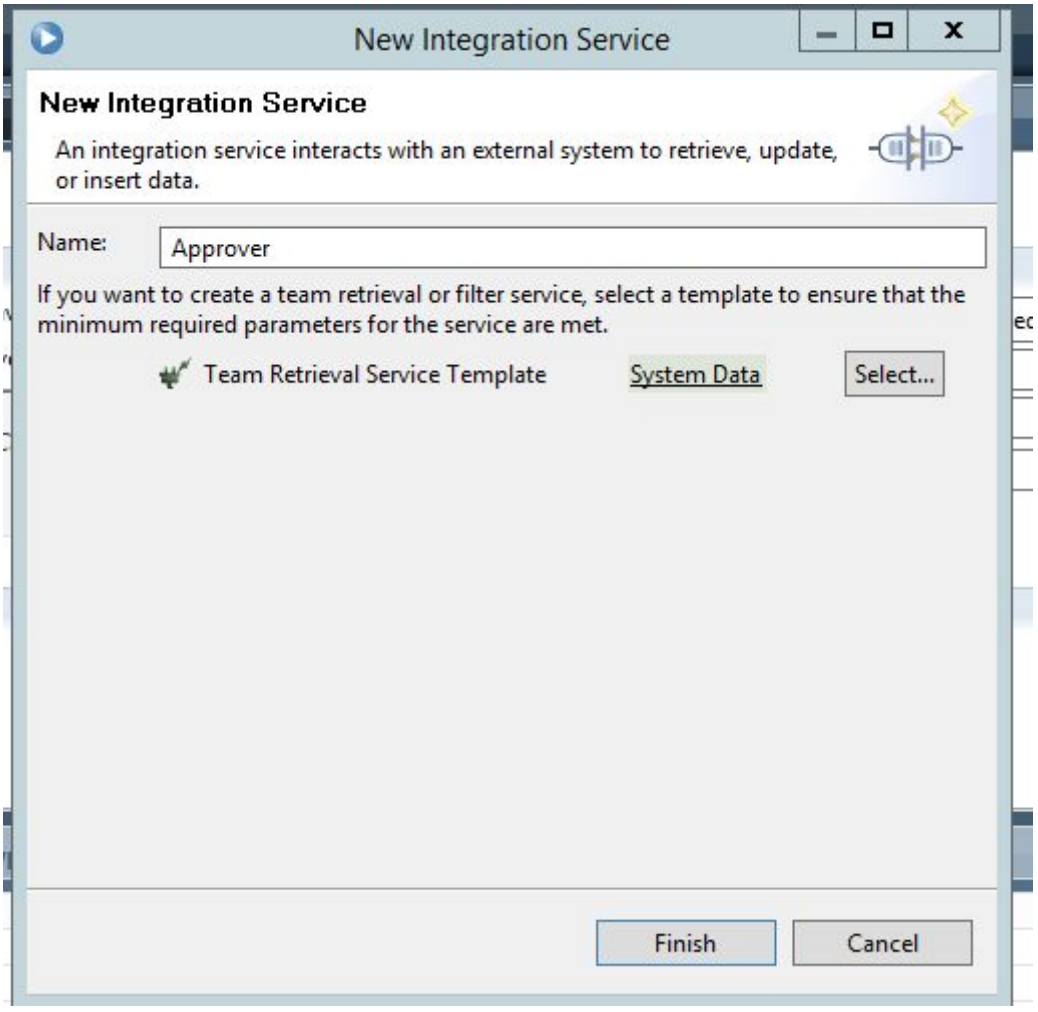
1.Form the Process Designer click on Teams, a pop window will show assign a name to the Team.



2.To create a Team using service we have to tick the specify team using service box.



3.To create a new Service click on new button a pop up window appears give the name and select the Team retrieval service template .



4.When you create Tean retrieval service you get name as the predefined input and Team as the output as shown in figure

FIg:Variables for Team Retrieval Service

.

Approach 1 : Creating a static team using Team Retrieval Service:

```
specify user and group names
var resultTeam = new tw.object.Team();
resultTeam.members = new tw.object.listOf.String();

resultTeam.members[0] = "David";
resultTeam.members[1] = "Linda";
resultTeam.members[2] = "Region North";

// provide the result
tw.local.team = resultTeam;
```

In this approach we create a static Team.If a new person needs to be added to the Team then we have to add him in the script and redeploy the application which is a costly process.So, we go for Dynamic way of creating the Team.

Approach 2: Creating a Dynamic Team using Team Retrieval Service

```
var resultTeam=new tw.object.Team();
var resultTeam1=new tw.object.Team();
resultTeam1= tw.system.org.findRoleByName("tw_allusers");
resultTeam.members=new tw.object.listOf.String();

for(var i=0;i<7;i++)
{
```

```
    resultTeam.members[i]=resultTeam1.allUsers[i].name;

}
tw.local.team=resultTeam;
```

I have created a group named tw_allusers from ProcessAdmin console and added users to the Team.In this service I am retrieving first seven numbers of the Team using Team Retrieval Service.

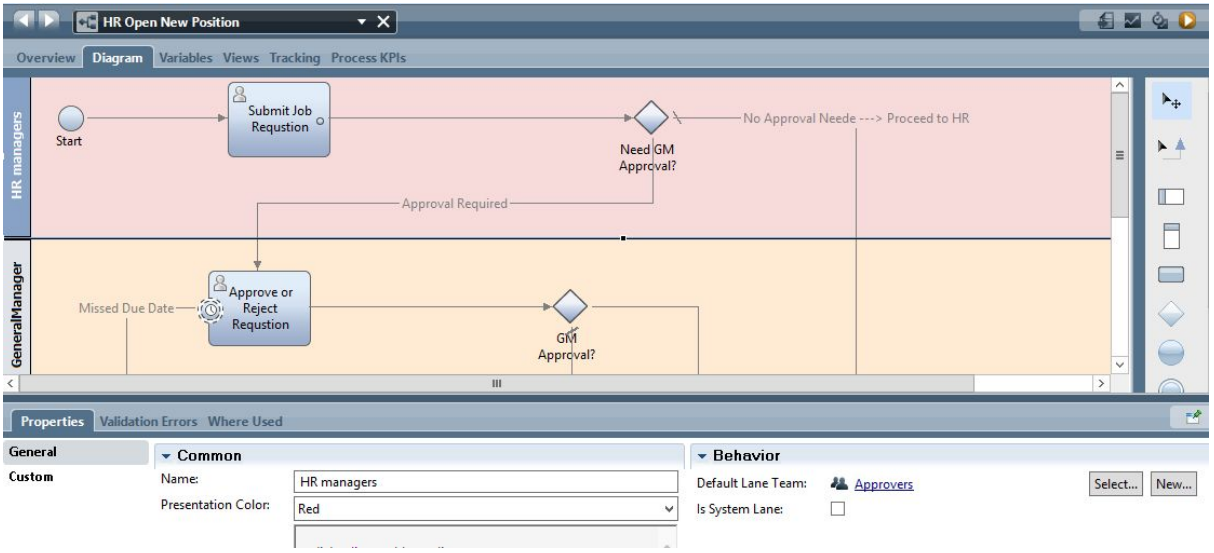


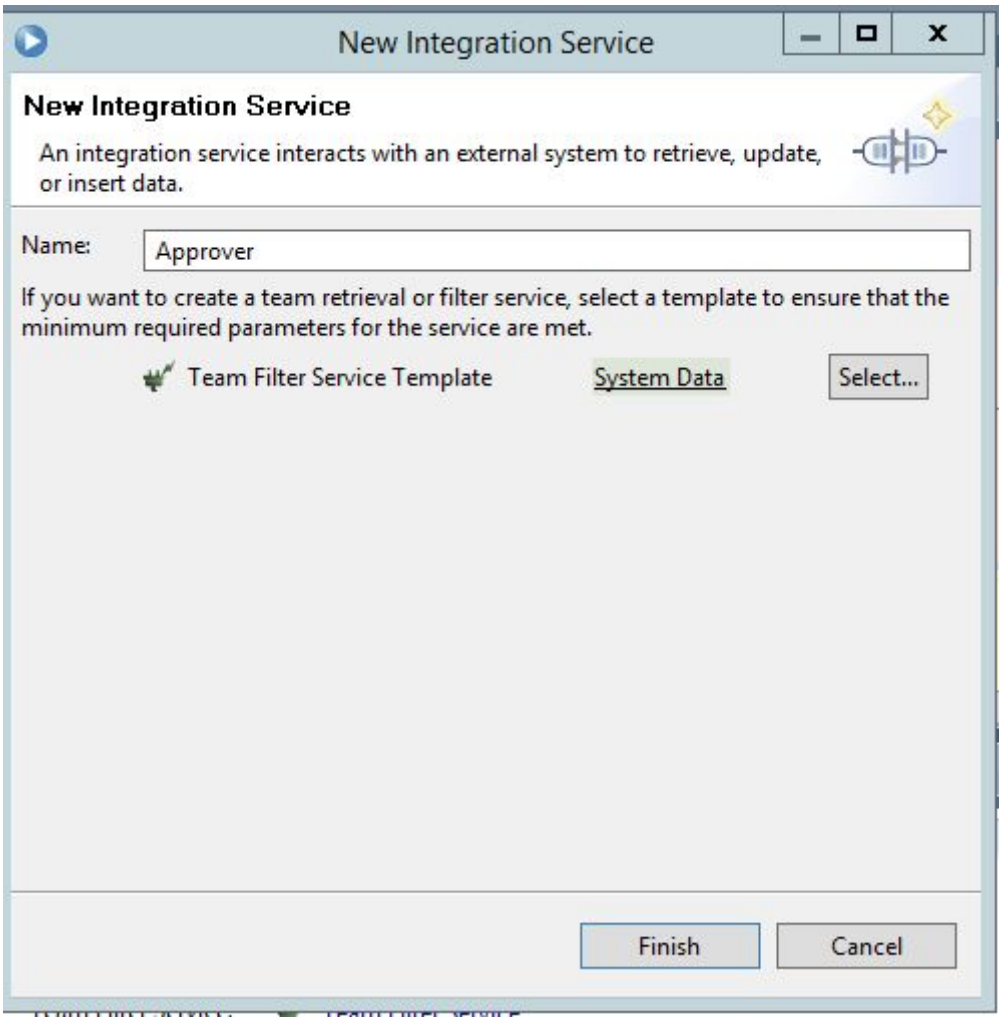
Fig:Adding Team Retrieval Service to the Lane

Team Filter Service: Team filter services are used when modeling human activities of a business process to customize the team that is used for people assignment at runtime You can use team filter services, for example, to filter out users that do not have the required skill level to work on a particular task, or to enable scenarios, such as the separation of duties.

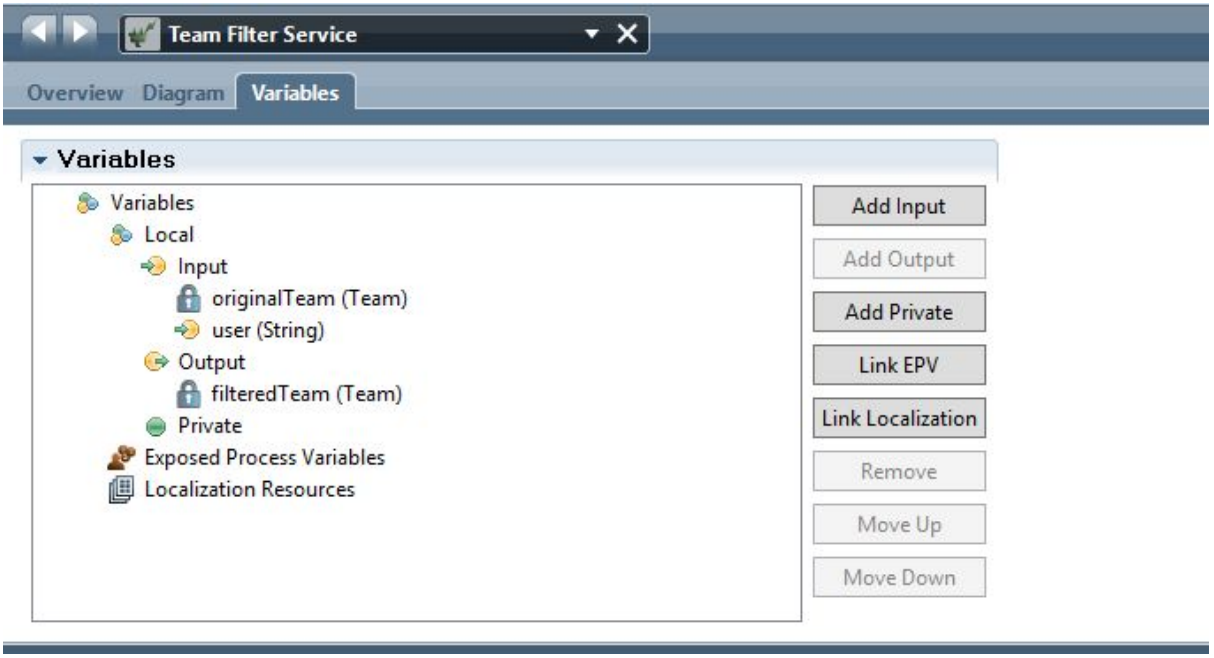
A team filter service is an implementation of an integration service that is based on the *Team Filter Service Template*. The template mandates that the service returns a *filteredTeam parameter* of predefined BO *Team* type and takes as input an *originalTeam* parameter of the BO *Team* type.

Creating Team Filter Service:

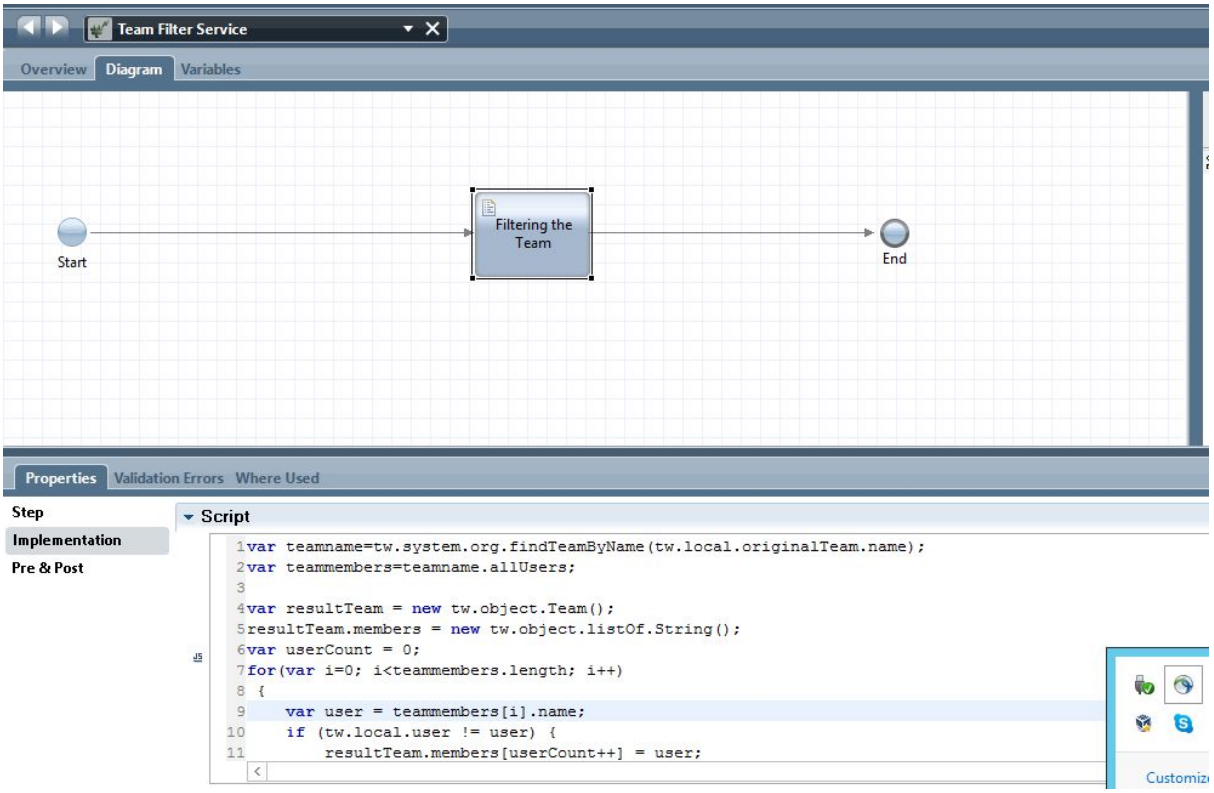
1. In the activity go to assignments property and click on new team filter service , a pop window appears give the name and select Team filter service template.



2.We get originalTeam as the input and filteredTeam as the output as shown in figure below.



3.Now take server script and write the logic to filter the Team as shown in the figure.



4.Script to filter the Team

```
var teamname=tw.system.org.findTeamByName(tw.local.originalTeam.name);
var teammembers=teamname.allUsers;

var resultTeam = new tw.object.Team();
resultTeam.members = new tw.object.listOf.String();
var userCount = 0;
for(var i=0; i<teammembers.length; i++)
 {
   var user = teammembers[i].name;
   if (tw.local.user != user) {
      resultTeam.members[userCount++] = user;
   }
 }
// set the result
tw.local.filteredTeam = resultTeam;
```

Here I am removing the team member from the team who has already claimed the task in the previous activity by getting the details from “tw.system.step.task.owner” script. In the previous activity I have assigned the owner of the task in post assignments to a local variable as shown in below figure.

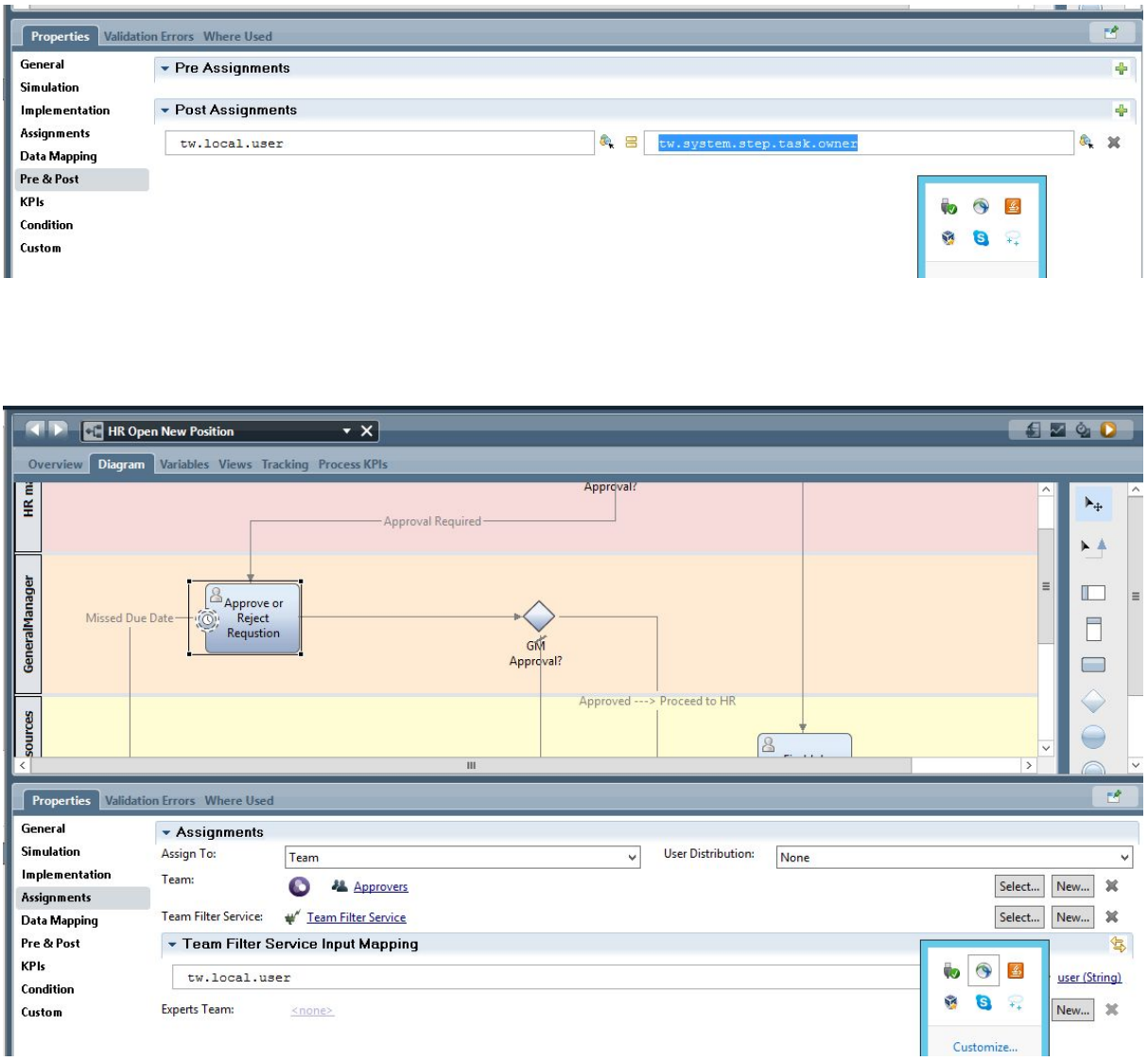


Fig:Assigning Team Filter Service to an Activity