

Classes 16 and 17

Learning Objectives

- Understand resource aspects of workflows
- Be able to model resource aspects in YAWL

Readings

This class is the third in a series of classes (13 to 20) which will introduce you to the control-flow, resource, and data-flow perspective of a workflow model. The previous classes covered the control-flow, that is concerned with defining the tasks that make up a workflow, and the order in which they are to be executed. In this class, we will define the resource perspectives for workflows.

Resources are typically people in the organization and the roles they play. The resource perspective is concerned with who is allowed to or who is supposed to carry out the tasks in a workflow.

As the previous classes, this class also relies on Chapters 2, 3, 8, and 10 of the textbook. We will try to guide your reading to focus on the resource perspective, but you will likely end up reading some material multiple times, as it is hard to really separate the control-flow, resource, and data perspectives. For this class, we will read some additional sections in Chapters 2 and 8 but also include parts of Chapter 10.

Once you have an understanding of the different resource patterns and the routing and interaction strategies supported by YAWL (Chapter 2), you will need to know how to actually use the YAWL software to implement them. This happens in two parts. In the YAWL control center web application, you can log into the YAWL resource service to define the organizational structure, i.e. the roles, positions, users, etc (Chapter 10). Once all this is defined, you can then assign resources to tasks in the YAWL editor (Chapter 8). Finally, you will learn to execute a workflow and observe it in action.

This class is heavily hands-on with the YAWL software. As for the previous classes, the YAWL users manual (installed on your computer with the software) might also be helpful for the detail. If you wish to read the user's manual, the relevant sections in there are 6.3.4, 6.3.5, 6.3.6, and 6.4.

You may also wish to have a look at the following tutorial videos on the course web-site:

- Video Y2: YAWL Control Center – Resource Definitions
- Video Y5: YAWL – Resource Allocation Demo
- Video Y6: YAWL – Process Execution Demo

Chapter 2

Chapter 2 is the core chapter of the textbook (and of the entire course) and you should read it carefully and understand it thoroughly. For this class, you should read the following sections that deal with resourcing and resource patterns:

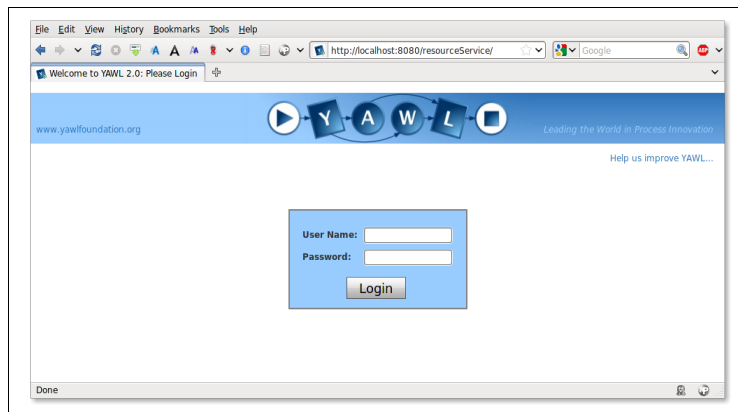
- Section 2.1
- Section 2.2.3
- Section 2.6 (**without** Figure 2.33 and 2.35)

Resources in the workflow context almost always means human resources. In fact, the two most important purposes of a workflow system is the routing of work items and interaction with the workflow users to make work items available (work items are *instances* of a task, e.g. task “Approve purchase order” is executed for purchase order 101, for purchase order 102, etc.). Section 2.2.3 introduces you to 7 different types resource patterns that the YAWL authors have observed in real workflows. While section 2.2.3 may be a little abstract and difficult to understand because it has no examples, you should be able to grasp the ideas behind the patterns. Things should become clearer once you read Section 2.6 which explains in detail how resource assignment is done in YAWL. Here, the organizational model is described (i.e. what are roles, positions, capabilities, etc.), the life-cycle of a work item from its creation to completion is described, along with all the actions that the user can do to a work item. You should have a good understanding of Figure 2.34, as the work item life-cycle is one of the central concepts in workflow management. The *interaction strategies* in Section 2.6.3 deal with how a work item gets to be worked on by a particular resource (person). The distinction between offering, allocating, and starting of a work item is important. Each of these three actions can be done either by the workflow system, or by a resource (person). The routing strategies in Section 2.6.4 deal with how possible resources for a task or work item are selected. You will see that this can be done based on roles, capabilities, and other criteria. Additional constraints can be imposed on the resources, such as the need to have a different person do a certain task, or the requirement to have the same person do different tasks. Finally, Section 2.6.5 gives a quick overview over the rights or privileges that users have with respect to what they are allowed to do with workflow items that have been allocated to them.

Chapter 10

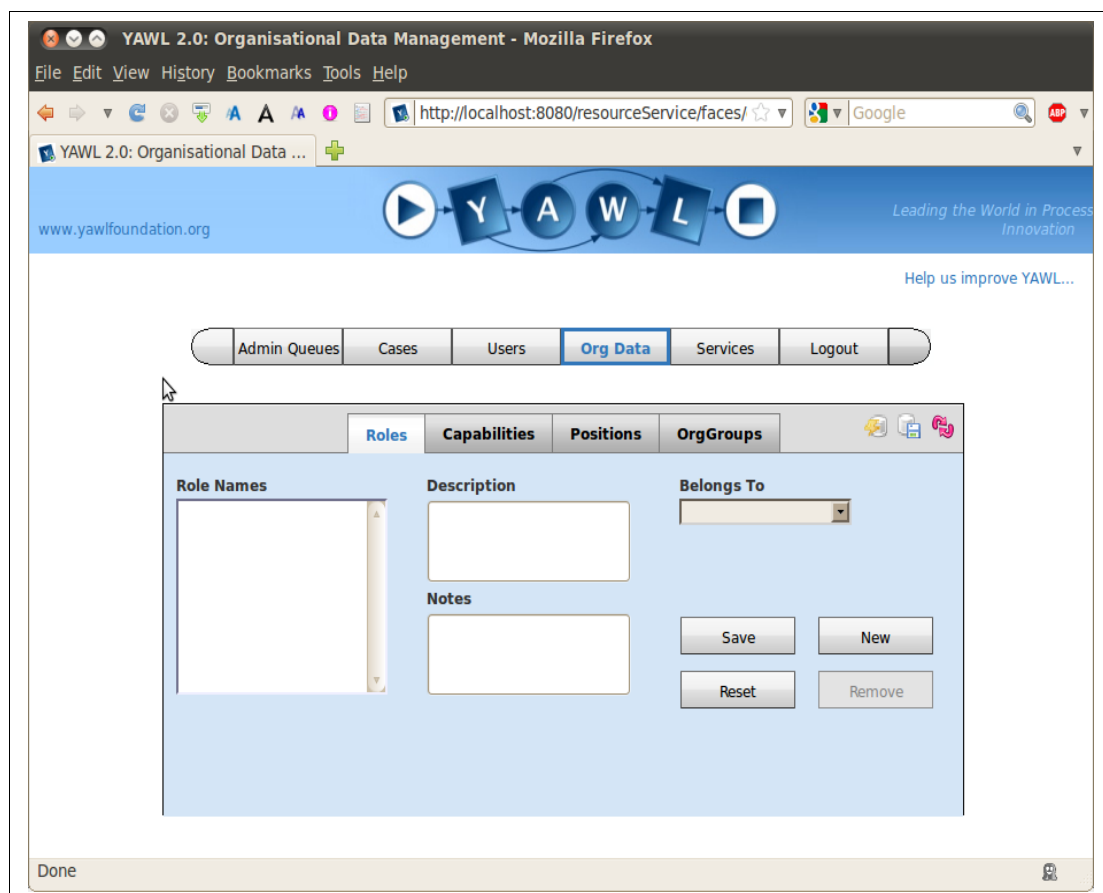
Now that you have an understanding of the different resource patterns and the routing and interaction strategies supported by YAWL, you will want to know how to actually use the YAWL software to implement them. This happens in two parts. In the YAWL control center web application, you can log into the YAWL resource service to define the organizational structure, i.e. the roles, positions, users, etc. Once all this is defined, you can then assign resources to tasks in the YAWL editor. Chapter 10 describes the resource service where you define the organizational model. You should read the entire chapter **except Section 10.4**.

If you have understood the principles of resource allocation and routing in Chapter 2 (Section 2.6), you will find that many of the sections in Chapter 10 are repetition with a bit of added detail. Section 10.2 provides another view over the interaction strategies in YAWL and Section 10.3 is again an introduction to the way that organizations are modelled in YAWL, this time with an example of what it looks like in the software. As I mentioned already, you will need to log as into the YAWL as an administrator.



YAWL login screen

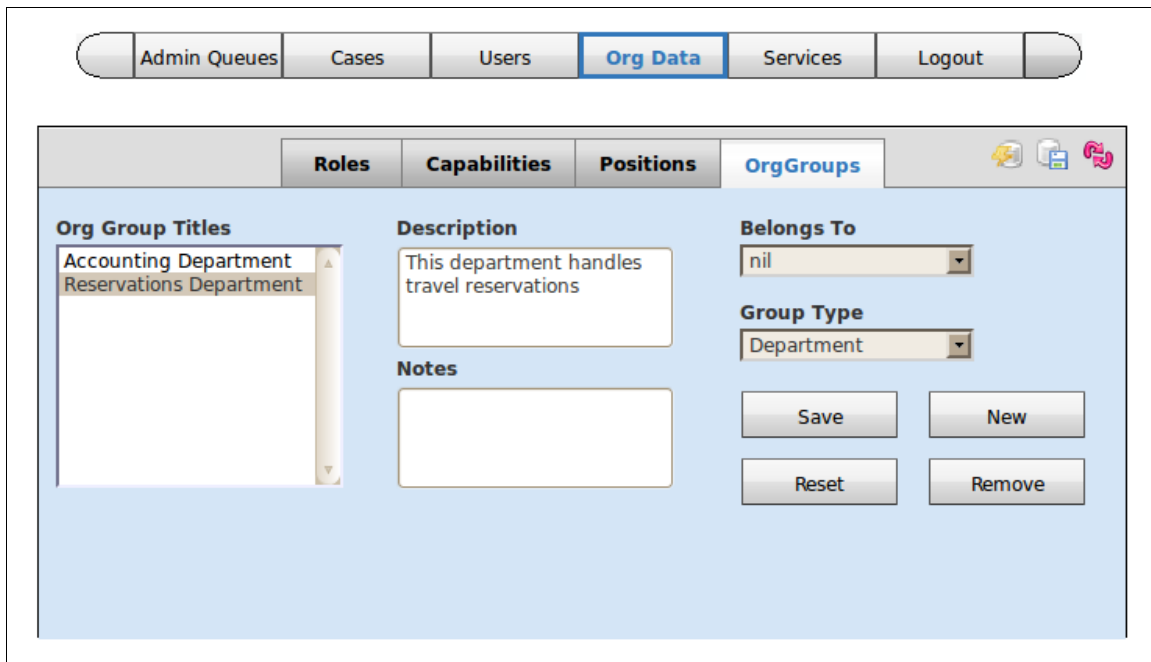
The first time, you can use the default user “admin” with password “YAWL”. As part of defining users and participants, you can create other users with administrator privileges (see Section 10.6 for details). Only administrative users can create and change the organizational data in YAWL, using the forms that are shown in Figures 10.3 and 10.3. The software interface itself is pretty self-explanatory once you have read and understood the concepts in Section 2.6 and Chapter 10. Once logged in, click on the “Org Data” tab:



Maintaining the organizational structure


You can now create and change users, roles, capabilities, positions, and organizational groups. It's probably best to first defined organizational groups, such as business units, departments, etc. The following screenshot shows two departments, the accounting department and the reservations

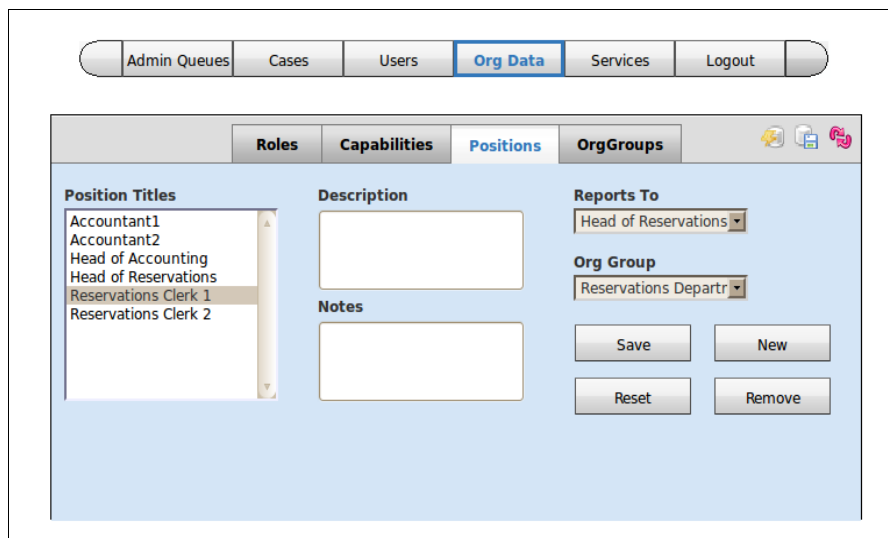
department. You can also specify that one unit belongs to another unit by selecting the unit it belongs to in the “belongs to” drop-down box (or leave it at “nil” if you don't want the unit to belong to another unit). The group type does not affect the behaviour of YAWL and is only for your own information.



The screenshot shows a web application interface with a top navigation bar containing 'Admin Queues', 'Cases', 'Users', 'Org Data' (highlighted), 'Services', and 'Logout'. Below this is a sub-navigation bar with 'Roles', 'Capabilities', 'Positions', and 'OrgGroups' (highlighted). The main content area is titled 'Org Groups' and contains three sections: 'Org Group Titles' with a list box showing 'Accounting Department' and 'Reservations Department'; 'Description' with a text area containing 'This department handles travel reservations'; and 'Notes' with an empty text area. To the right of these sections are two dropdown menus: 'Belongs To' (set to 'nil') and 'Group Type' (set to 'Department'). At the bottom right are four buttons: 'Save', 'New', 'Reset', and 'Remove'. A 'refresh' icon is visible in the top right corner of the form area.

Maintaining organizational groups

Once you have created your organizational units, you can define positions that belong to these groups, e.g. “Secretary”, “Forlift driver” or whatever else you're modeling. You can assign each position to an organizational group in the “Org Group” drop-down box, and also define reporting lines, using the “Reports to” drop-down box. In the example below, I have specified three positions for the accounting department and three positions for the reservations department. Tip: Sometimes you need to refresh the data in the drop-down boxes by clicking the “refresh” icon  in the top right of the form.



The screenshot shows the same web application interface, but with the 'Positions' tab selected in the sub-navigation bar. The main content area is titled 'Positions' and contains three sections: 'Position Titles' with a list box showing 'Accountant1', 'Accountant2', 'Head of Accounting', 'Head of Reservations', 'Reservations Clerk 1', and 'Reservations Clerk 2'; 'Description' with an empty text area; and 'Notes' with an empty text area. To the right of these sections are two dropdown menus: 'Reports To' (set to 'Head of Reservations') and 'Org Group' (set to 'Reservations Depart'). At the bottom right are four buttons: 'Save', 'New', 'Reset', and 'Remove'. A 'refresh' icon is visible in the top right corner of the form area.

Maintaining positions and reporting lines

Finally, you can define roles and capabilities, as in the two screenshots below, where I have defined two capabilities and four roles.

The screenshot shows a web application interface for maintaining capabilities. At the top, there is a navigation bar with four tabs: 'Roles', 'Capabilities' (which is selected and highlighted in blue), 'Positions', and 'OrgGroups'. To the right of the tabs are three small icons: a yellow lightning bolt, a blue document, and a red circular arrow. The main content area has a light blue background. On the left, there is a list box titled 'Capability Names' containing two items: 'Authorized to book flights' and 'Authorized to handle cash'. To the right of the list box are two text input fields: 'Description' and 'Notes'. At the bottom right, there are four buttons: 'Save', 'New', 'Reset', and 'Remove'.

Maintaining capabilities

The screenshot shows a web application interface for maintaining roles. At the top, there is a navigation bar with four tabs: 'Roles' (which is selected and highlighted in blue), 'Capabilities', 'Positions', and 'OrgGroups'. To the right of the tabs are three small icons: a yellow lightning bolt, a blue document, and a red circular arrow. The main content area has a light blue background. On the left, there is a list box titled 'Role Names' containing four items: 'Accounts payable clerk', 'Accounts receivable clerk', 'Business Manager', and 'Reservations agent'. To the right of the list box are two text input fields: 'Description' and 'Notes'. To the right of the 'Description' field is a dropdown menu titled 'Belongs To' with the value 'nil' selected. At the bottom right, there are four buttons: 'Save', 'New', 'Reset', and 'Remove'.

Maintaining roles

Once you have set up the organizational structure, positions, capabilities and roles that you need, you can add users (people). These are the people that will log in to YAWL and execute the workflow items that you define. As you add new users, you can assign them to roles, positions, and capabilities. You can also define privileges (described in detail in Section 10.6) and select whether those users are administrators of the system (i.e. whether they can add other users and change the organizational data). An example of a new user is shown in the screenshot below.

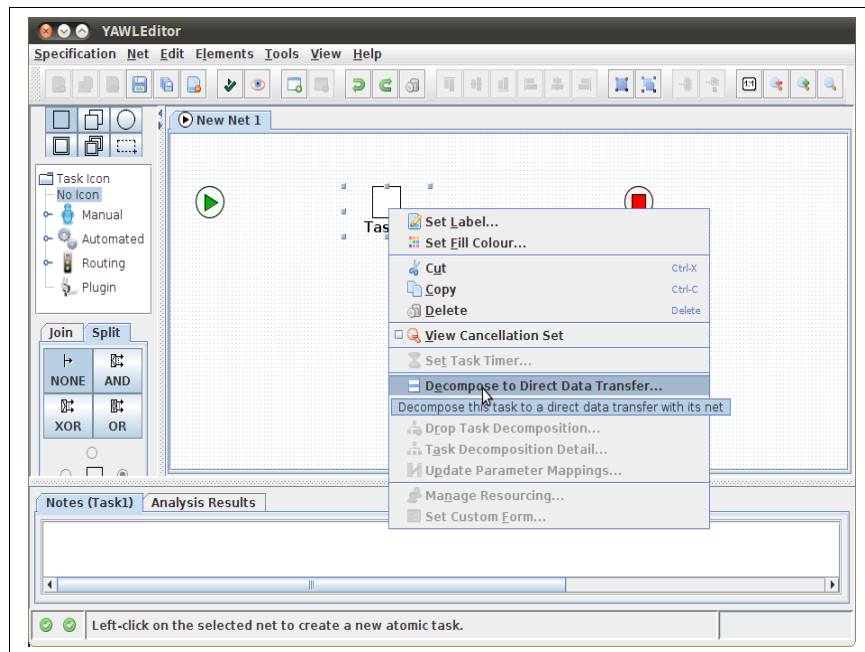
Adding a new user

Chapter 8

Once the organizational structure and the users are defined in the resource service, they can be used in workflow models in the editor. When you start the editor, it will automatically try to connect to the resource service you have configured (you have done this as part of the previous class, check those notes). Thus, the editor “knows” all about your organizational structures and Section 8.4 shows you how you can select the resource allocation and interaction strategies in the editor. For this class you should read

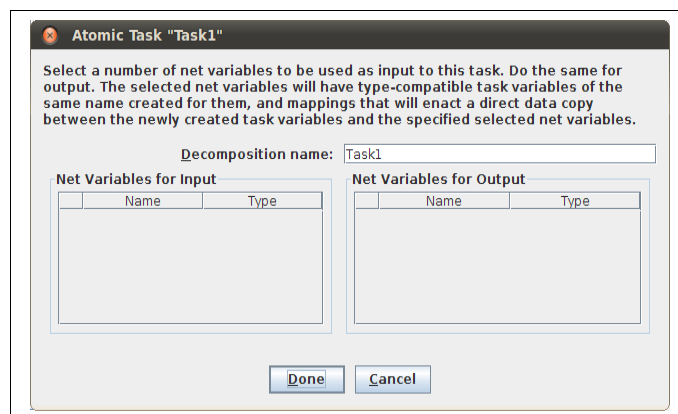
- Section 8.4

In order to assign resources to a task in the editor, you need to first “decompose” the task to a “Direct data transfer”. At the moment about the terminology, this will become clear in the next class when we talk about data in a workflow. You can do this by right-clicking on the task and selecting the appropriate option in the menu, see below:



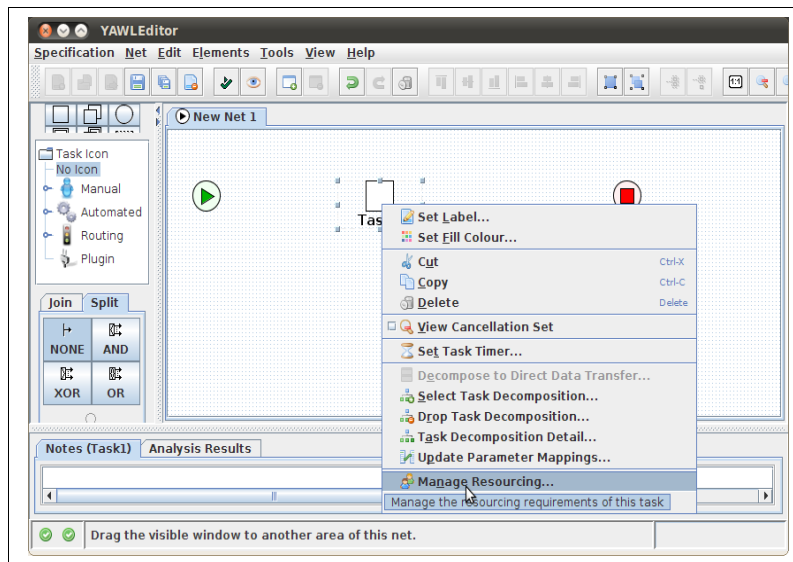
Decomposing a task to Direct Data Transfer

In the following dialog box, you can define the input and output data for the task. For now, just click “Done”. We will come back to this in the next class when we define how data flows through the tasks in a workflow:



Defining data input and output for a task

Once you have done this, you can manage the resourcing for a task. Again, right-click on the task and this time choose “Manage resourcing” from the menu:



Managing the resourcing

If you get an error message that says there are no roles or positions defined in the resource service, the editor has probably not connected to the resource service. You should have set up this connection when you installed YAWL in the previous class, so go back to those notes and double-check.

If all goes well, you will see a series of dialog boxes as shown and explained in Chapter 8 (Figures 8.11 to 8.15) where you can define the resources for each task.

Once you have finished defining the workflow specification and resourcing in the editor, it is time to actually execute the specification. For this, it needs to be loaded into the workflow engine through the YAWL web-interface. Choose the “Cases” tab, browse to the YAWL file that you have saved on your computer and upload the file to the YAWL workflow engine. YAWL checks the specification for errors and will tell you if there's something wrong. If all goes well, the specification you have just uploaded should appear in the list of “Loaded specifications”. You can select it, and then click “Launch case” to start an instance of the workflow. This instance will then appear in the list of “Running cases”.

Admin Queues **Cases** Users Org Data Services Logout

Upload Specification

/home/joerg/Desktop/test1.yawl Browse...

Upload File

Loaded Specifications

| | | |
|------|-----|--------------------------------|
| test | 0.1 | No description has been given. |
|------|-----|--------------------------------|

Launch Case Unload Spec

Running Cases

Cancel Case

Uploading specifications and launching cases

Depending on your resourcing, the tasks might be automatically assigned or offered to users. Alternatively, they may be waiting for manual assignment or offering (described in Section 10.5 in Chapter 10). If you are logged in as an administrator, you can check the “Admin Queues” to see tasks that are unoffered. You can then offer, allocate or start the work items. You can also see tasks that are worklisted, i.e. already offered or allocated to resources. You can reoffer, reallocate or restart these items with different resources (users).

Admin Queue Cases Users Org Data Services Logout

Unoffered (0) **Worklisted (1)**

Work Items

2.1:Task1_3

Specification test **Task** Task1 Reoffer

Case 2.1 **Status** Executing Reallocate

Created Sep:11, 2010 11:48:20 **Age** 0:00:02:55 Restart

Assigned To (1) evermann, joerg **Resource State** Started

Admin Queues for unoffered and worklisted items

As a user, you also have access to the worklist. This is the queue of work items that have been offered or allocated to you or that you have started (described in section 10.7 in Chapter 10). Different options are available to you depending on the status of the work item.

The screenshot displays a web application interface for managing work queues. At the top, there is a navigation bar with buttons: 'Work Queue' (highlighted), 'Edit Profile', 'Team Queues', 'Cases', and 'Logout'. Below this, a sub-navigation bar shows four tabs: 'Offered (0)', 'Allocated (0)', 'Started (1)', and 'Suspended (0)'. The 'Started (1)' tab is active. The main content area is divided into three sections. On the left, under the heading 'Work Items', there is a list box containing the item '2.1:Task1_3'. In the center, under the heading 'Specification', there are three fields: 'test' under 'Specification', '2.1' under 'Case', and 'Sep:11, 2010 11:48:20' under 'Created'. To the right of these, under the heading 'Task', there are two fields: 'Task1' under 'Task' and 'Executing' under 'Status'. Below the 'Status' field, there is an 'Age' field showing '0:00:09:23'. On the far right, there is a vertical stack of buttons: 'View/Edit' (disabled), 'Suspend', 'Reallocate s/l', 'Reallocate s/f', 'New Instance', and 'Complete'.

Work Queues

You will notice that as you execute the work items, there is not much to see. In particular, you cannot view or edit the work item; the button in the screenshot above is inactive/grayed out. This is because there is not yet any data associated with this workflow and the work items that you could view or edit. We will work on this in the next class.

Once you complete a work item that has been started, the next work item in the workflow will be created and offered or allocated to the appropriate resource. For this class, just make sure you understand how the resource allocation and interaction strategies work.

Review Questions


At the end of this chapter, you should be able to answer the following questions:

- Describe the life-cycle of a work item, i.e. the states it can be in, and the actions that the system or users can take in each state
- Describe the general organization model, i.e. what are the relationships between roles, capabilities, positions, organizational groups and users?
- What is the difference between a position and a role?
- Name and explain the 7 types of resource patterns and describe at least three examples of each
- What is the difference between chained execution and piled execution?
- Describe the three main interaction in YAWL with respect to resource assignment.
- Name and explain the 8 different work item interaction strategies in YAWL.
- Name and explain the 6 different work item routing strategies in YAWL.
- What is the difference between round robin (by time), round robin (by least frequency) and round robin (by experience) allocation?
- Explain the difference between user privileges and task privileges in YAWL.

Review Exercises

- Chapter 2, Exercises 7, and 8.
- Chapter 10, Exercises 1, 2, 3, 4, 5, 7

The order fulfilment process for exercise 5 is the example that comes with YAWL. Choose YAWL4Study -> Misc -> Examples on your computer. In the |orderfulfillment” folder, you will see the workflow specification for the editor (“orderfulfillment.yawl”) and the organizational data that you can import

into the YAWL resource service (“orderfulfillment.ybcp”). To import this, choose the import icon  in the “Org Data” tab. The form will change to allow you to upload this file.

Group work tip: Each of you can then log in as one of the users defined in the order fulfilment process from wherever you are and work through a realistic routing and interaction scenario.

- Chapter 8, Exercise 5 (focus on the resource aspect)

The following exercises complement those in Chapter 2 which you may have done as review exercises for the last class.

- For Chapter 2, Exercise 2:

The travel agency consists of two department: reservations and accounting. John Doe manages the reservations department and Jane Bloggs manages the accounting department. Anna South and Michelle North work in reservations and report to John, while Jim East and Bob West work in accounting and report to Jane. Only Jane and Jim have the authority to handle cash from customers, while only John is able to make flight reservations, while John, Anna and Michelle can all book hotels and cars. Jim used to work in reservations, so he can still make hotel and car bookings if needed, but he should not process the payments for any bookings that he makes.

Hotel, car and flight reservations are of course made by people in the reservations department, while the payment activities are done by people in accounting. To make things easier for the customer, hotel, car, and flight reservations should be done by the same person, if that person is available. To speed up the payment processing, it should be done to the person who is currently least busy.

Implement the organizational structure in the YAWL resource service by creating appropriate organizational units, roles, positions, capabilities, and users. Then, alter you model (from the previous class) to include the resource assignment. Upload the model to the YAWL resource engine and create a small number of workflow instances. You will need to log in and out as different users (pretending you are different people) to execute the workflow. Make sure the execution matches your expectation based on the interaction and routing strategies you have modelled. If not, identify where you have gone wrong.

Group work tip:

Instead of using Jane Bloggs, Anna South, Michelle North, John Doe, Jim East and Bob West, you can use your group members' names and define those on the YAWL system. Each of you can then log in from wherever you are and work through a realistic routing and interaction scenario.

- For Chapter 2, Exercise 3:

The organizational structure and the positions are not really important, but there are two roles. Employees can work as staff members to submit improvement recommendations. Employees can also work as improvement recommendation managers. However, no staff member should manage the recommendation that they themselves submitted. Requests for further information should obviously be given to those staff members that submitted the request. The detailed examination must not be done by the same manager that handled the initial recommendation. Also, the confirmation must not be done by the same manager who has done the detailed examination. The archiving task can be done by any staff member, except the one who submitted the initial recommendation. Given these constraints, ensure an overall rapid response to the improvement recommendations.

Note: you can add organizational structures, positions and more roles if it helps you to enforce the constraints.

Implement the organizational structure in the YAWL resource service by creating appropriate organizational units, roles, positions, capabilities, and users. Then, alter you model (from the previous class) to include the resource assignment. Upload the model to the YAWL resource engine and create a small number of workflow instances. You will need to log in and out as different users (pretending you are different people) to execute the workflow. Make sure the execution matches your expectation based on the interaction and routing strategies you have modelled. If not, identify where you have gone wrong.

Group work tip:

You can use your group members' names and define those on the YAWL system. Each of you can then log in from wherever you are and work through a realistic routing and interaction scenario.