

CamilleX Documentation

None

Thai Son Hoang

None

Table of contents

1. CamilleX User Manual	3
2. Getting Started	4
2.1 Installation	4
2.2 Configuration	4
2.3 IMPORTANT	4
2.4 Basic Tutorial	5

1. CamilleX User Manual

CamilleX new constructs (called `XMachines` and `XContexts`) for `Event-B` modelling. The new constructs are text files which are automatically translated into the corresponding `Rodin`'s `Event-B` constructs (i.e., `Machines` and `Contexts`) accordingly. Facility for translating to and from `Rodin`'s components to `CamilleX` components can be invoked manually. `CamilleX` is inspired by `Camille` text editor for `Rodin` and is based on `XText` technology, hence the name `CamilleX`.

- *Getting Started:*
- *Installation:* Information for installing the *CamilleX* feature.
- *Basic tutorial:* This tutorial provides a step-by-step walk-through working with `CamilleX` constructs.

2. Getting Started

2.1 Installation

[CamilleX](#) is available from the main Rodin update site (under [CamilleX](#) category). There are two versions of the feature, the standard version for users and the SDK version for software developers which include source code.

2.2 Configuration

Windows users must change the workspace text file encoding to *UTF-8*. This can be updated under the [Rodin Preferences](#) [General/Workspace](#) then in the [Text file encoding](#) section, select Other: [UTF-8](#).

2.3 IMPORTANT

Currently, *CamilleX* not only supports *standard* Event-B machines and contexts, but also supports *Machine Inclusion* (for composition), and *Record* extension to the Event-B modelling language.

Since the *XContexts* and *XMachines* are compiled to the Rodin files, the corresponding Rodin contexts and machines will be **OVER-WRITTEN**. Any changes in the Rodin files will not be lost.

DO NOT USE the *CamilleX* if you use modelling plug-ins that use the Rodin files as source such as *UML-B* state-machines and class-diagrams, as the additional modelling elements will be over-written.

2.4 Basic Tutorial

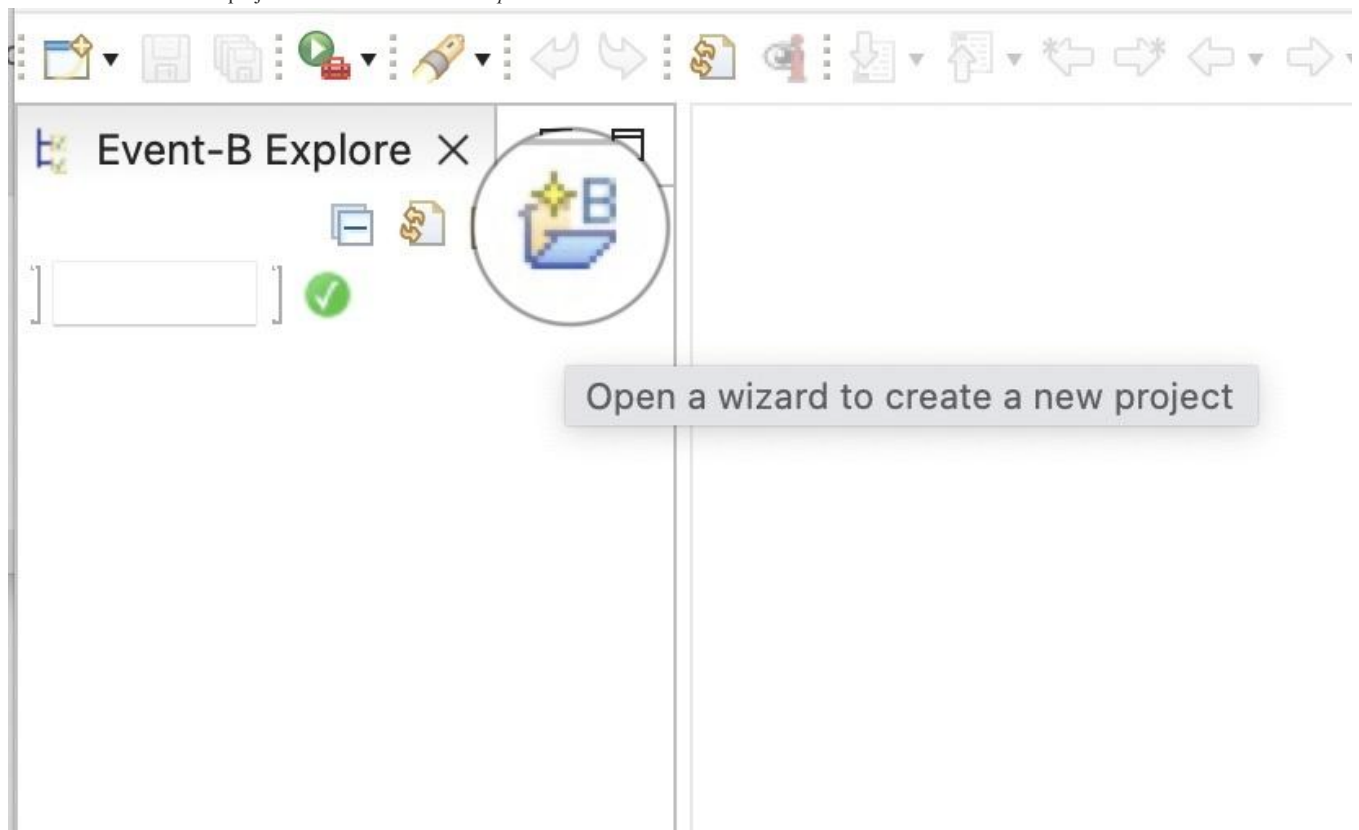
2.4.1 Task 1. Create an Event-B Project

Introduction

The purpose of this task is to create an Event-B project for the CamilleX constructs.

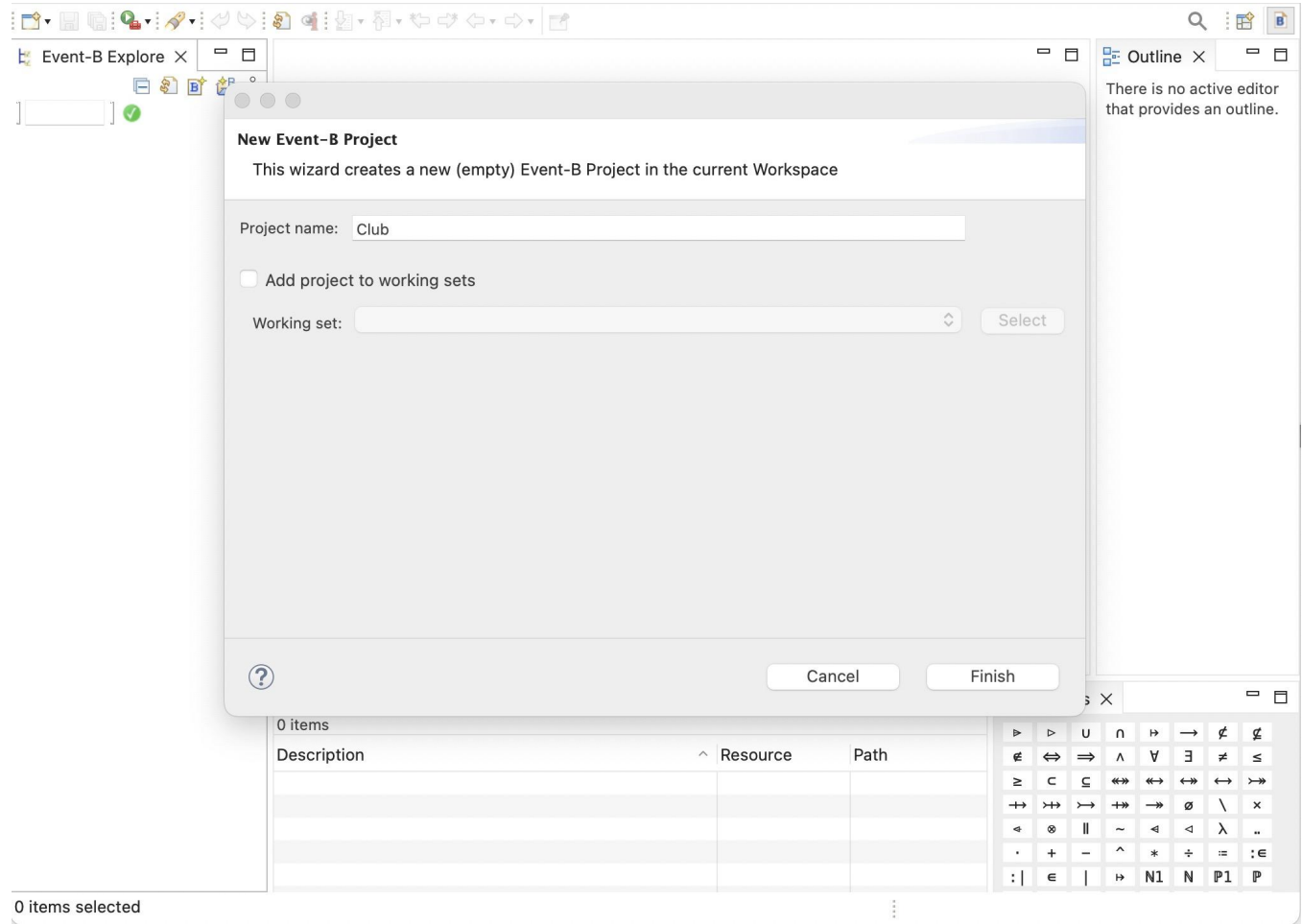
Step 1. Create a New Event-B Project Named `Club`

- Click on the new Event-B project button on the *Event-B Explorer*.

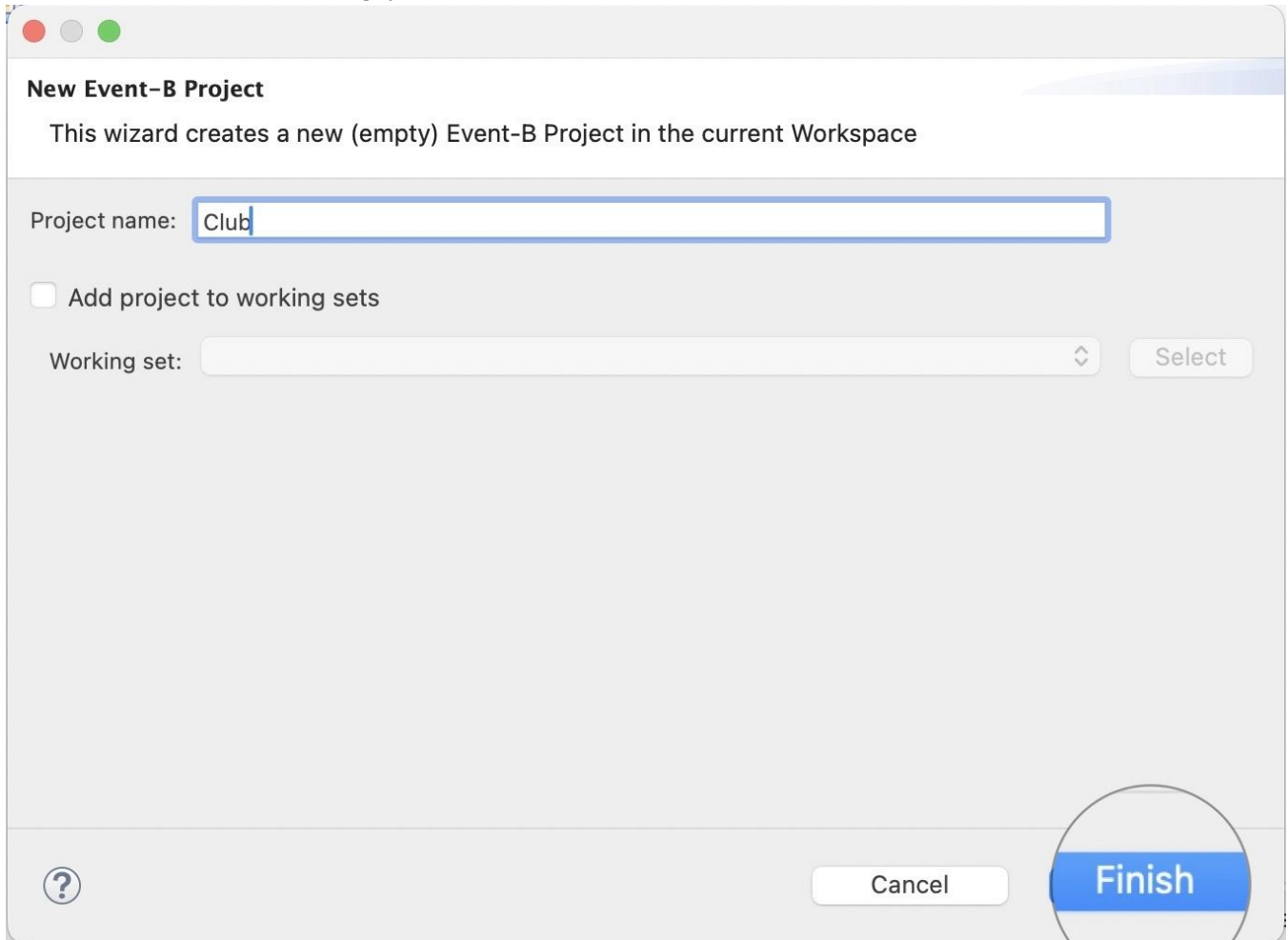


(The same wizard can be invoke through the menu `File -> New -> Event-B Project`.)

- From the pop-up dialog, enter `Club` as the `Project name`



- Click **Finish** to confirm the creation of the project.



New Event-B Project

This wizard creates a new (empty) Event-B Project in the current Workspace

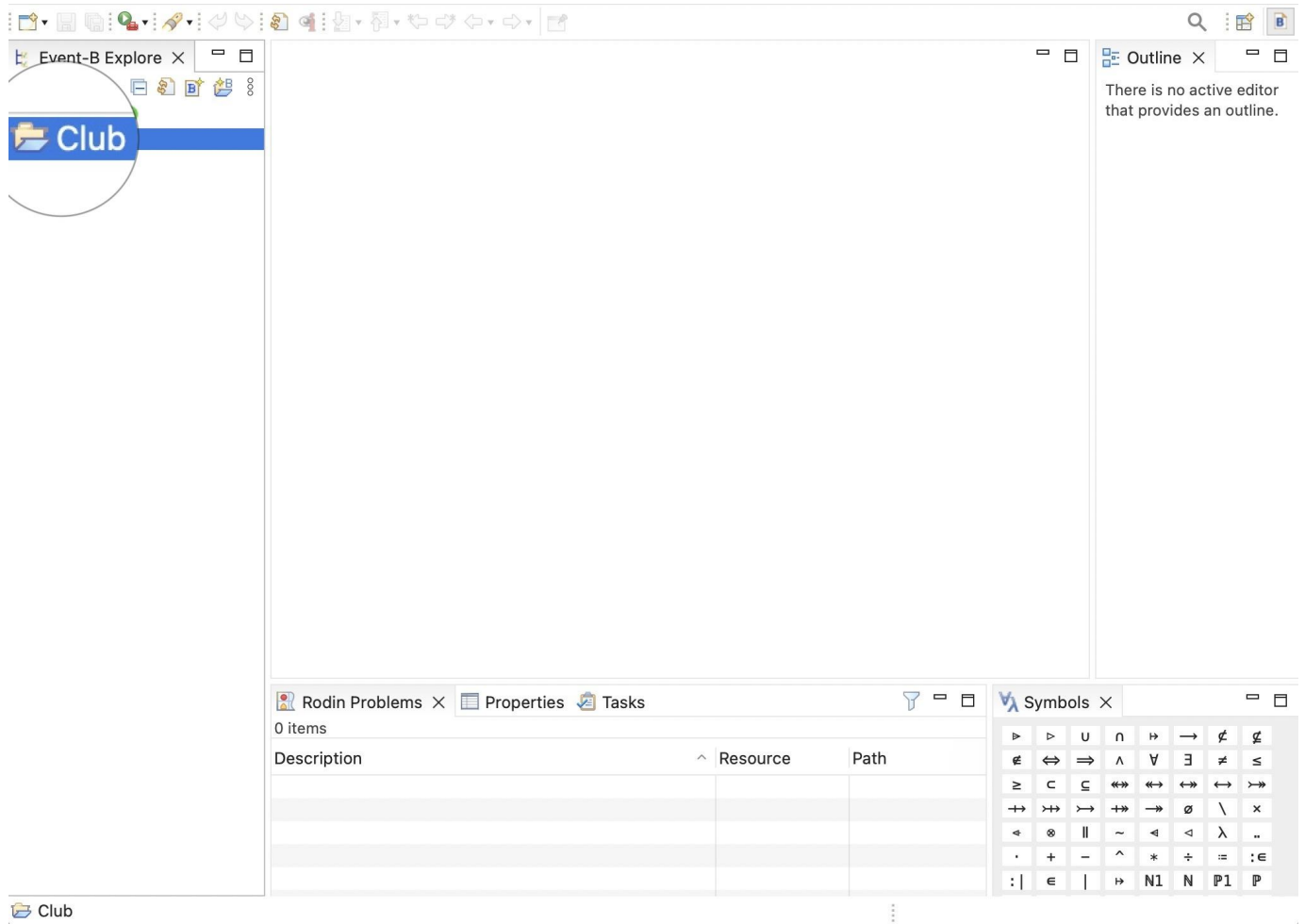
Project name:

☐ Add project to working sets

Working set:

Conclusion

By now, the project `Club` should be visible in the *Event-B Explorer*.



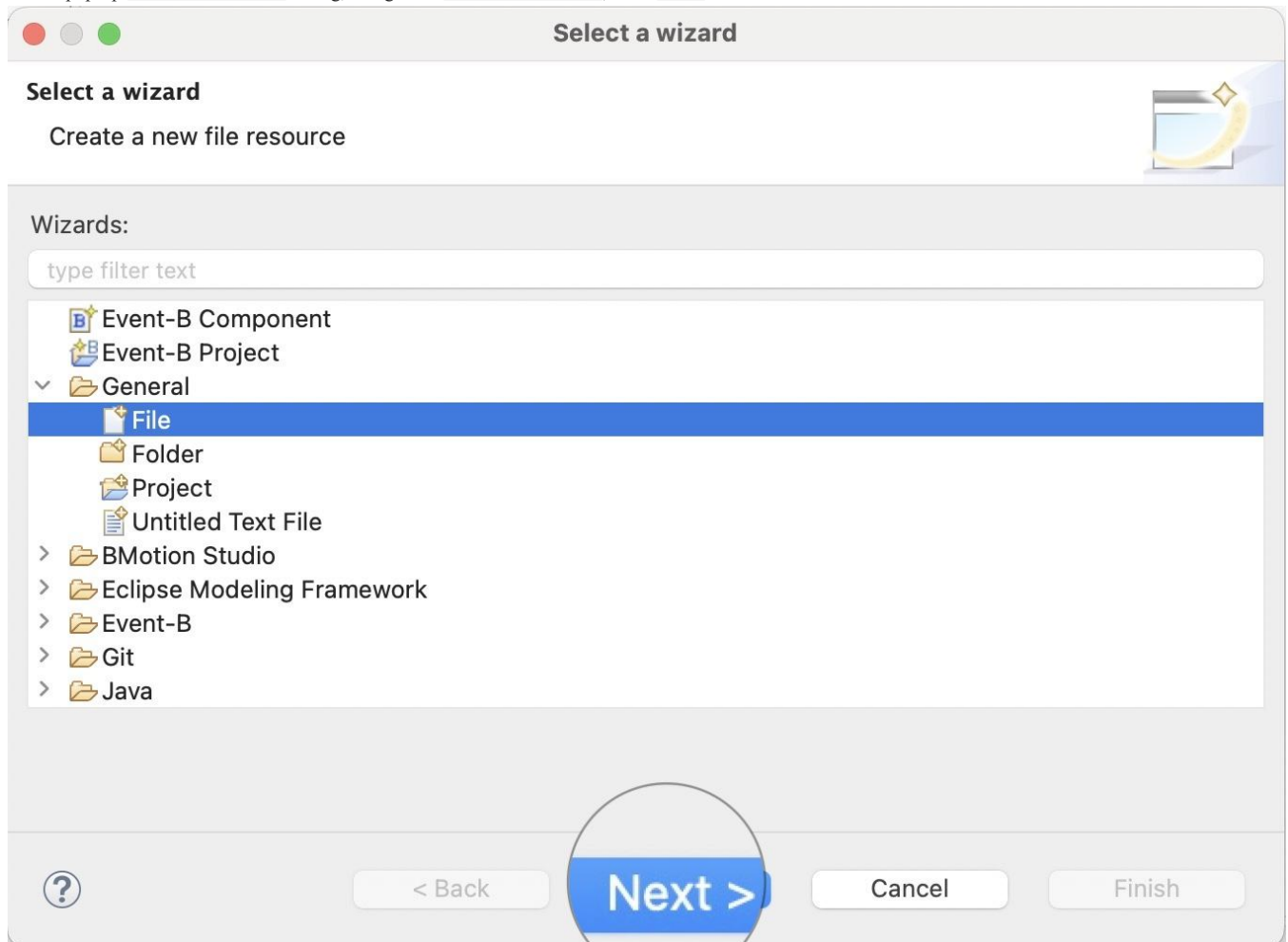
2.4.2 Task 2. Create an XContext

Introduction

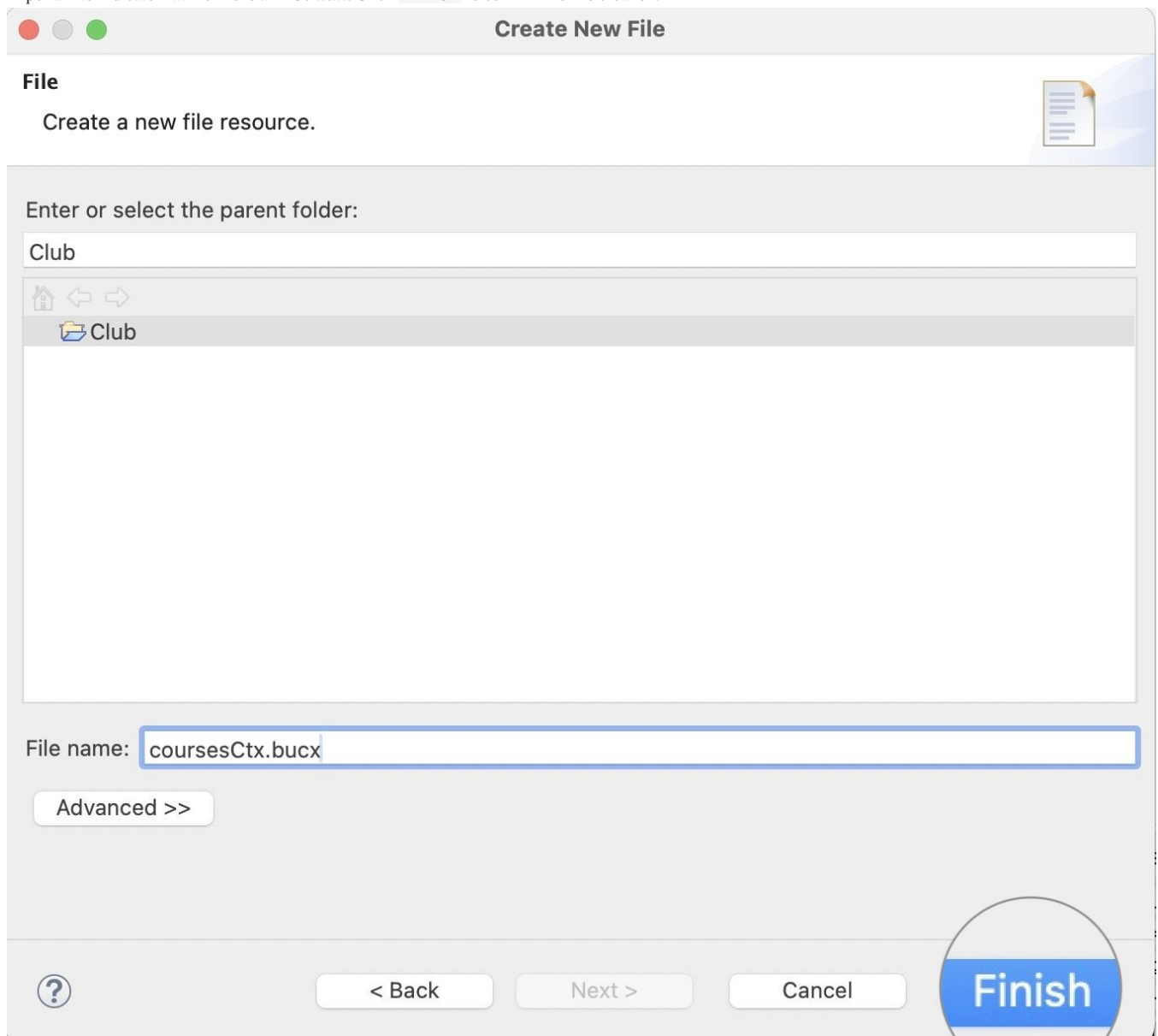
The purpose of this task is to create a simple XContext within the newly created project.

Step 1. Create a New XContext Named `coursesCtx.bucx`

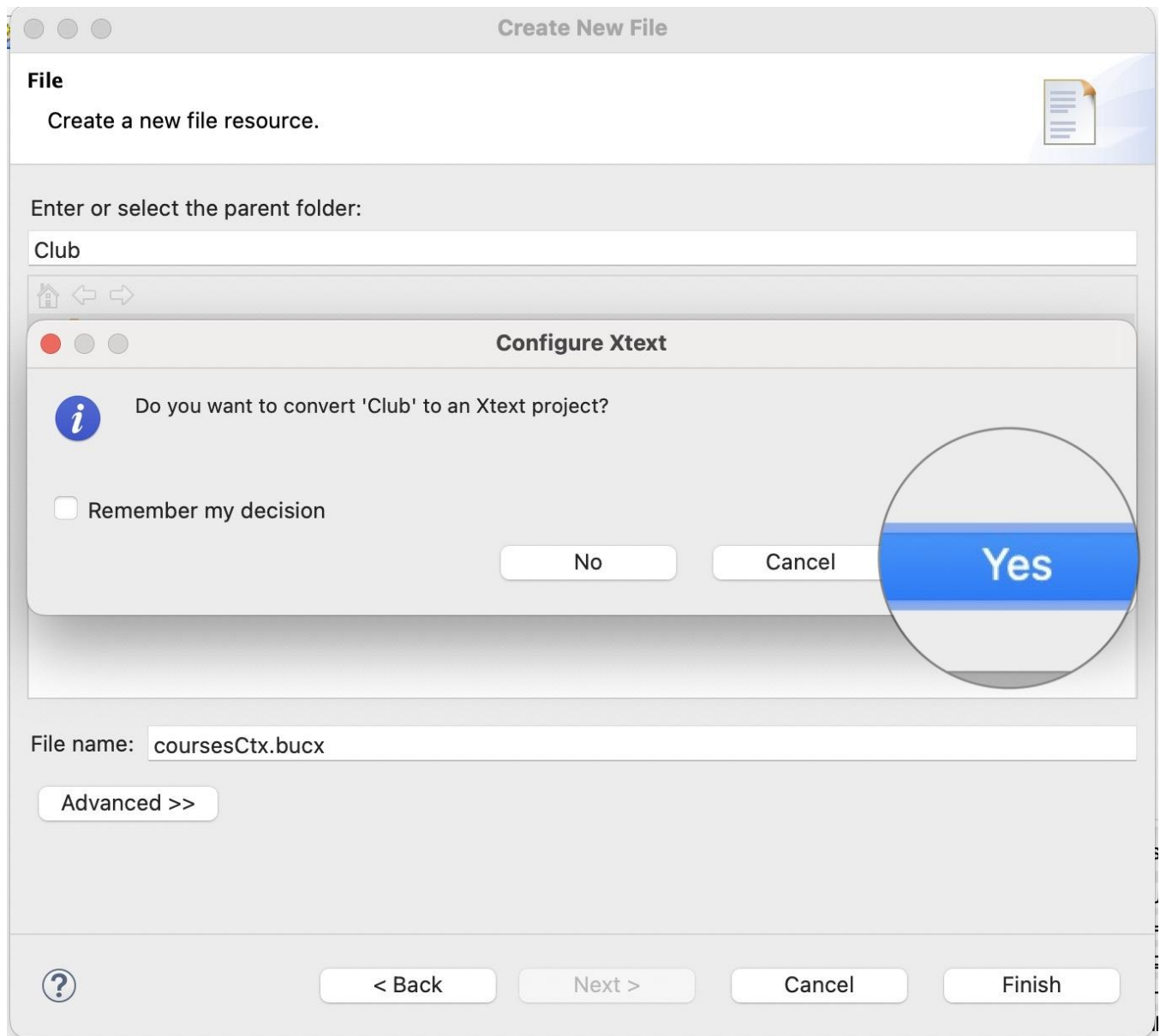
- Use the menu `File -> New -> Other` to open the `Select a wizard` dialog.
- On the pop-up `Select a wizard` dialog, navigate to `General -> File`, click `Next`.



- On the `Create New File` dialog, choose `Club` project as the parent folder, and put `coursesCtx.bucx` as the `File name`. The file extension `.bucx` is important to indicate that the file is an *XContext*. Click `Finish` to confirm the file creation.



- Important:** A pop-up dialog will be displayed asking to convert the `Club` project to an *XText* project, please answer **Yes**. This enables the *XText* builder to work automatically for converting CamilleX constructs to Rodin constructs.



(If you miss this step, you can invoke it via right click on the `Club` project from the *Event-B Explorer* and `Configure -> Convert to XText Project`). The new created file `coursesCtx.bucx` will be opened automatically in an editor. It has some error markers and we will fix this in the next step.

Step 2. Set the Content of `courseCtx.bucx`

- Using the editor, set the content of `coursesCtx.bucx` as follows.

```
context coursesCtx
sets
  CRS    // The set of all courses
constants
  m      // The maximum number of courses
axioms
  @axm0_1: finite(CRS) // There can only be a finite number of courses
  @axm0_2: m ∈ N1     // The maximum number of courses is a non-zero natural number
theorem @thm0_1: 0 < m // The maximum number of courses is positive
end
```

```
*coursesCtx.bucx X
context coursesCtx
sets
  CRS    // The set of all courses
constants
  m      // The maximum number of courses
axioms
  @axm0_1: finite(CRS)    // There can only be a finite number of courses
  @axm0_2: m ∈ ℕ1        // The maximum number of courses is a non-zero natural number
theorem @thm0_1: 0 < m    // The maximum number of courses is positive
end
```