

Modelling Software-based Systems

Dominique Méry
LORIA & Telecom Nancy
Université de Lorraine
<https://members.loria.fr/Mery>
dominique.mery@loria.fr

November 23, 2025(10:31 P.M.)

This repository contains course notes, exercises, models and projects from three courses given as part of master's level training on the Event-B formal method. It provides access to resources in the form of pdf files or Rodin archives. The relationship with the Atelier-B platform is explained.

The table of contents shows the summary of three main courses (at Université de Lorraine/University of Lorraine) based on our experiment using the Event-B method:

- The first course **MCFSI** is part of the curriculum of the last year students of Telecom Nancy who are focusing on software engineering.
- The second course **MsC in Computer Science** is aimed at students on the IT Masters course at the University of Lorraine.
- The third course **DISCONT** is aimed at Telecom Nancy students in their final year who are specialising in software engineering and embedded systems. The course focuses on hybrid systems and hybrid models. The work of the ANR DISCONT project (see website at ANR or website at LORIA)
- The fourth course **MsC in Computer Science** is aimed at students on the IT Masters course at the Maynooth University and is a special cooking for students of my colleague Professor Rosemary Monahan, who is welcoming me every year. This course is a special edition that is shorter than the first two courses, which is great news! The aim is to give students the specific elements of the Event-B language and, above all, to encourage them to follow a few tutorials.

The Event-B method is based on a modelling language used to describe state-based models and safety properties of those state-based models. The originality of Event-B lies in its ability to enable incremental and proof-based modelling of *reactive systems*. The Event-B language contains both set notations and a first-order predicate calculus; it offers the possibility of defining models of reactive systems called machines and contexts and includes the refinement relationship that allows us to follow an incremental development methodology.

Contents

2	Course MCFSI at Telecom Nancy	4
2.1	Slides for the course MCFSI	4
2.1.1	Lecture 1 The Modelling Language Event-B.	4
2.1.2	Lecture 2 Proof Obligations.	4
2.1.3	Lecture 3 Checking contracts using Event-B.	4
2.1.4	Lecture 4 Correctness by Construction with the Modelling Language Event-B using the Refinement.	4
2.1.5	Lecture 5 Access Control	4
2.2	Tutorials	4
2.3	Project	4
2.4	Event-B Models	4
2.4.1	Event-B Archives for the lectures	5
2.4.2	Event-B Archives for the tutorial 1	5
2.4.3	Event-B Archives for the tutorial 2	5
2.4.4	Event-B Archives for the tutorial 3	5
2.4.5	Event-B Archives for the tutorial 4	5
2.4.6	Event-B Archives	6
2.5	Past exams of the course MCFSI	6
3	Course MsC in Computer Science: Modelling and verifying software-based systems for Master in Computer Science of the University of Lorraine	6
3.1	Slides of the course	6
3.1.1	Lecture 1 The Modelling Language Event-B.	6
3.1.2	Lecture 2 Proof Obligations.	6
3.1.3	Lecture 3 Correctness by Construction with the Modelling Language Event-B using the Refinement.	6
3.1.4	Lecture 4 Access Control	6
3.1.5	Lecture 5 Checking contracts with Event-B	6
3.2	Tutorials	6
3.3	Event-B Models	6
3.3.1	Event-B Archives for the lectures	7
3.3.2	Event-B Archives for the tutorial 1	7
3.3.3	Event-B Archives for the tutorial 2	7
3.3.4	Event-B Archives for the tutorial 3	7
3.4	Project	8
3.5	Past exams of the Course MsC in Computer Science: Modelling and verifying software-based systems for Master in Comp	8
4	Course DISCONT: Modelling hybrid systems	8
4.1	Documentation	8
4.2	Slides for the course 2020-2021	8
4.3	Slides for the course 2021-2022	8
5	Course EMU Event-B at Maynooth University	9
5.1	Slides for the course EMU	9
5.1.1	Lecture 1 The Modelling Language Event-B.	9
5.1.2	Lecture 2 Proof Obligations.	9
5.1.3	Lecture 3 Checking contracts using Event-B.	9

5.1.4	Lecture 4 Correctness by Construction with the Modelling Language Event-B using the Refinement.	9
5.1.5	Tutorial 1 and 2 Tutorial Notes.	9
5.1.6	Lecture 5 Access Control	9
5.2	Tutorials	9
5.3	Event-B Models	9
5.3.1	Event-B Archives for the lectures	9
5.3.2	Event-B Archives for the tutorials	9
6	Course MsC in Computer Science 2024/2025 MsC: Modelling and verifying software-based systems for Master in Computer Science of the University of Lorraine	10
6.1	Slides of the course	10
6.1.1	Lecture 1 The Modelling Language Event-B.	10
6.1.2	Lecture 2 Proof Obligations.	10
6.1.3	Lecture 3 Correctness by Construction with the Modelling Language Event-B using the Refinement.	10
6.1.4	Lecture 4 Access Control	10
6.1.5	Lecture 5 Checking contracts with Event-B	10
6.2	Tutorials	10
6.3	Event-B Models	10
6.3.1	Event-B Archives for the lectures	10
6.3.2	Event-B Archives for the tutorial 1	10
6.3.3	Event-B Archives for the tutorial 2	11
6.3.4	Event-B Archives for the tutorial 3	11
6.4	Project	11
6.5	Past exams of the Course MsC in Computer Science: Modelling and verifying software-based systems for Master in Comp	11

1 Documentation and Tools

The main document on the Event-B modelling language and methodology is the book of Jean-Raymond Abrial [1].

The Rodin platform is available at the following link.

The list of symbols is useful for typing symbols which will appear as mathematical notations.

The first chapter on the modelling language Event-B chapter 1 summarizes details on the foundations and on the applications.

The second chapter on using the modelling language Event-B for developing sequential algorithms chapter 2 with examples.

The third chapter on using the modelling language Event-B for verifying contracts for sequential algorithms chapter 3 with examples.

The chapter [3] (from [2].) authored by Dominique Cansell and Dominique Méry and entitled *The Event-B Modelling Method: Concepts and Case Studies* has been published

from lectures notes given in a Summer School and you can use it for getting details from Event-B see the following link.

Lectures Notes *The Modelling Language* at the following link.

Chapter *Event-B* at the following link.

2 Course MCFSI at Telecom Nancy

2.1 Slides for the course MCFSI

2.1.1 Lecture 1 The Modelling Language Event-B.

2.1.2 Lecture 2 Proof Obligations.

2.1.3 Lecture 3 Checking contracts using Event-B.

2.1.4 Lecture 4 Correctness by Construction with the Modelling Language Event-B using the Refinement.

2.1.5 Lecture 5 Access Control

2.2 Tutorials

Tutorial 1 Using the Event-B modelling language on simple examples.

Tutorial 2 Using the refinement in the Event-B modelling language for deriving sequential algorithms.

Tutorial 3 Using the refinement in the Event-B modelling language for deriving systems.

Tutorial 4 Using the Event-B modelling language for verifying contracts.

2.3 Project

The assessment of students is based on two works:

- A first project on verification which is stated in the following document *Projet Vérification 3A IL* and the deadline is November 15, 2025.
- A second project which is stated in the following document *Projet Correction par construction 3A IL* and the deadline for the project is February 1, 2026 and the presentation is organised on Monday, February 10, 2026.

2.4 Event-B Models

The Event-B models related to the tutorials are given in the next list:

2.4.1 Event-B Archives for the lectures

Archive Rodin for explaining differences between Event-B invariant and Event-B theorem.

Archive Rodin for the management of school.

Archive Rodin for clock modelling.

Archive Rodin for modelling the design of the factorial function.

Archive Rodin for modelling the design of the factorial function.

2.4.2 Event-B Archives for the tutorial 1

Archive Rodin mcfsi1-ex1-tut1.zip.

Archive Rodin mcfsi1-ex2-tut1.zip.

Archive Rodin mcfsi1-simple.zip.

Archive Rodin mcfsi1-variant1.zip.

Archive Rodin mcfsi1-variant2.zip.

Archive Rodin mcfsi1-summation.zip.

Archive Rodin mcfsi1-ressource-pb1.zip.

Archive Rodin mcfsi1-ressource-pb2.zip.

Archive Rodin mcfsi1-invariantsafety.zip.

Archive Rodin mcfsi1-ex8.zip.

Archive Rodin ex8-tut1.zip.

Archive Rodin mcfsi1-ex9.zip.

Archive Rodin mcfsi1-ex10.zip.

2.4.3 Event-B Archives for the tutorial 2

Archive Rodin alg-maxtwonumbers.zip.

Archive Rodin alg-ex1.zip.

Archive Rodin alg-ex2.zip.

Archive Rodin alg-ex3.zip.

Archive Rodin alg-ex4.zip.

Archive Rodin alg-simple.zip.

2.4.4 Event-B Archives for the tutorial 3

Archive Rodin mcfsi3-ex1-plugin.zip.

Archive Rodin mcfsi3-ex2.zip.

Archive Rodin mcfsi3-ex2.zip.

Fichier power2.c.

Fichier power2.h.

Archive Rodin mcfsi3-ex3.zip.

Archive Rodin mcfsi3-ex4.zip.

Archive Rodin mcfsi3-ex5.zip.

2.4.5 Event-B Archives for the tutorial 4

Archive Rodin mcfsi4-ex1.zip.

Archive Rodin mcfsi4-ex2.zip.

Archive Rodin mcfsi4-ex3.zip.

Archive Rodin mcfsi4-ex4.zip.

2.4.6 Event-B Archives

Archive Rodin mrg1.zip.

Archive Rodin abacus.zip.

2.5 Past exams of the course MCFSI

Exam 2017

Exam 2018

Exam 2019

Exam 2020

Exam 2021

Exam 2022

Exam 2023

3 Course MsC in Computer Science: Modelling and verifying software-based systems for Master in Computer Science of the University of Lorraine

3.1 Slides of the course

3.1.1 Lecture 1 The Modelling Language Event-B.

3.1.2 Lecture 2 Proof Obligations.

3.1.3 Lecture 3 Correctness by Construction with the Modelling Language Event-B using the Refinement.

3.1.4 Lecture 4 Access Control

3.1.5 Lecture 5 Checking contracts with Event-B

3.2 Tutorials

Tutorial 1 Using the Event-B modelling language.

Tutorial 2 Designing and verifying sequential algorithms using the Event-B modelling language.

Tutorial 3 Modelling systems using the Event-B.

Tutorial 4 Using Event-B for verifying sequential annotated algorithms.

Tutorial 5 Still refinement

3.3 Event-B Models

The Event-B models related to the tutorials are given in the next list:

3.3.1 Event-B Archives for the lectures

Archive Rodin for explaining differences between Event-B invariant and Event-B theorem.

Archive Rodin for the access control system.

Archive Rodin for the management of school.

Archive Rodin for the factorial function.

3.3.2 Event-B Archives for the tutorial 1

Archive Rodin ex1-tut1.zip.

Archive Rodin ex2-tut1.zip.

Archive Rodin ex4-tut1.zip.

Archive Rodin ex51-tut1.zip.

Archive Rodin ex52-tut1.zip.

Archive Rodin ex6-tut1.zip.

Archive Rodin ex7-tut1.zip.

Archive Rodin ex8-tut1.zip.

Archive Rodin ex9-tut1.zip.

Archive Rodin mcfsi1-variant1.zip.

Archive Rodin mcfsi1-variant2.zip.

Archive Rodin ex10-1-tut1.zip.

Archive Rodin ex10-2-tut1.zip.

Archive Rodin ex11-tut1.zip.

3.3.3 Event-B Archives for the tutorial 2

Exercise 1 Tutorial 2 We give two solutions using the variable ok or not.

Archive Rodin fx1-tut2.zip.

Archive Rodin fx1-tut2bis.zip.

Exercise 2 Tutorial 2 Archive Rodin fx2-tut2.zip.

Exercise 3 Tutorial 2 Archive Rodin mcfsi3-ex3.zip.

Exercise 4 Tutorial 2 Archive Rodin fx4-tut2.zip.

Exercise 5 Tutorial 2 Archive Rodin fx5-tut2.zip.

Exercise 6 Tutorial 2 Archive Rodin fx6-tut2.zip.

3.3.4 Event-B Archives for the tutorial 3

Exercise 1 Tutorial 3 Nous donnons deux solutions possibles: l'une avec ok et l'autre sans ok.

Archive Rodin ggx1-tut3.zip.

Exercise 2 Tutorial 3 Archive Rodin ggx2-tut3.zip.

3.4 Project

The assessment of students is based on two works:

- A written exam 1h30
- A project which is stated in the following document [Projet Master](#)

The deadline for the project is February 15, 2026 and the presentation will be organised later in February 2026.

3.5 Past exams of the Course MsC in Computer Science: Modelling and verifying software-based systems for Master in Comp

Archiuives sous forme d'une archive ZIP

4 Course DISCONT: Modelling hybrid systems

4.1 Documentation

The report entitled *A Refinement Strategy for Hybrid System Design with Safety Constraints* is giving details on the method Event-B in hybrid strategy.

The report entitled *From System Events to Software Operations for Refinement-based Modeling of Hybrid Systems* studies the integration of both approaches *system* and *software*.

A Static Checker for Reference Tracking Systems via Laplace Transform and Transfer Functions.

4.2 Slides for the course 2020-2021

Lecture 1

Lecture 2

4.3 Slides for the course 2021-2022

Lecture 1

Lecture 2

Lecture 3

Lecture 4

5 Course EMU Event-B at Maynooth University

5.1 Slides for the course EMU

5.1.1 Lecture 1 The Modelling Language Event-B.

5.1.2 Lecture 2 Proof Obligations.

5.1.3 Lecture 3 Checking contracts using Event-B.

5.1.4 Lecture 4 Correctness by Construction with the Modelling Language Event-B using the Refinement.

5.1.5 Tutorial 1 and 2 Tutorial Notes.

5.1.6 Lecture 5 Access Control

5.2 Tutorials

Tutorial 1 Using the Event-B modelling language on simple examples.

Tutorial 2 Designing and verifying sequential algorithms using the Event-B modelling language

5.3 Event-B Models

The Event-B models related to the tutorials are given in the next list:

5.3.1 Event-B Archives for the lectures

Archive Rodin for explaining differences between Event-B invariant and Event-B theorem.

Archive Rodin for the management of school.

Archive Rodin for clock modelling.

Archive Rodin for modelling the design of the factorial function.

5.3.2 Event-B Archives for the tutorials

Archive Rodin maynoothmodels.tar.gz.

6 Course MsC in Computer Science 2024/2025MsC: Modelling and verifying software-based systems for Master in Computer Science of the University of Lorraine

6.1 Slides of the course

6.1.1 Lecture 1 The Modelling Language Event-B.

6.1.2 Lecture 2 Proof Obligations.

6.1.3 Lecture 3 Correctness by Construction with the Modelling Language Event-B using the Refinement.

6.1.4 Lecture 4 Access Control

6.1.5 Lecture 5 Checking contracts with Event-B

6.2 Tutorials

Tutorial 1 Using the Event-B modelling language.

Tutorial 2 Designing and verifying sequential algorithms using the Event-B modelling language.

Tutorial 3 Modelling systems using the Event-B.

Tutorial 4 Using Event-B for verifying sequential annotated algorithms.

Tutorial 5 Still refinement

6.3 Event-B Models

The Event-B models related to the tutorials are given in the next list:

6.3.1 Event-B Archives for the lectures

Archive Rodin for explaining differences between Event-B invariant and Event-B theorem.

Archive Rodin for the management of school.

6.3.2 Event-B Archives for the tutorial 1

Archive Rodin ex1-tut1.zip.

Archive Rodin ex2-tut1.zip.

Archive Rodin ex4-tut1.zip.

Archive Rodin ex51-tut1.zip.

Archive Rodin ex52-tut1.zip.

Archive Rodin ex6-tut1.zip.

Archive Rodin ex7-tut1.zip.

Archive Rodin ex8-tut1.zip.

Archive Rodin ex9-tut1.zip.

Archive Rodin mcfsi1-variant1.zip.

Archive Rodin mcfsi1-variant2.zip.

Archive Rodin ex10-1-tut1.zip.
Archive Rodin ex10-2-tut1.zip.

6.3.3 Event-B Archives for the tutorial 2

Exercice 1 Tutorial 2 Nous donnons deux solutions possibles: l'une avec ok et l'autre sans ok.

Archive Rodin fx1-tut2.zip.
Archive Rodin fx1-tut2bis.zip.

Exercice 2 Tutorial 2 Archive Rodin fx2-tut2.zip.

Exercice 3 Tutorial 2 Archive Rodin mcfsi3-ex3.zip.

Exercice 4 Tutorial 2 Archive Rodin fx4-tut2.zip.

Exercice 5 Tutorial 2 Archive Rodin fx5-tut2.zip.

Exercice 6 Tutorial 2 Archive Rodin fx6-tut2.zip.

6.3.4 Event-B Archives for the tutorial 3

Exercice 1 Tutorial 3 Nous donnons deux solutions possibles: l'une avec ok et l'autre sans ok.

Archive Rodin ggx1-tut3.zip.

Exercice 2 Tutorial 3 Archive Rodin ggx2-tut3.zip.

6.4 Project

The assessment of students is based on two works:

- A written exam 1h30
- A project which is stated in the following document [Projet Master](#)

The deadline for the project is February 15, 2025 and the presentation will be organised later in February 2025.

6.5 Past exams of the Course MsC in Computer Science: Modelling and verifying software-based systems for Master in Comp

Archives sous forme d'une archive ZIP

References

- [1] J.-R. Abrial. *Modeling in Event-B: System and Software Engineering*. 2010.
- [2] Dines Bjørner and Martin C. Henson, editors. *Logics of Specification Languages*. EATCS Textbook in Computer Science. Springer, 2007.
- [3] Dominique Cansell and Dominique Méry. *The event-B Modelling Method: Concepts and Case Studies*, pages 33–140. Springer, 2007. See [2].