



Software Analyzers

## OVERVIEW OF THE FRAMA-C BOOK

---

Nikolai Kosmatov<sup>1</sup>

Virgile Prevosto<sup>2</sup>

Julien Signoles<sup>2</sup>

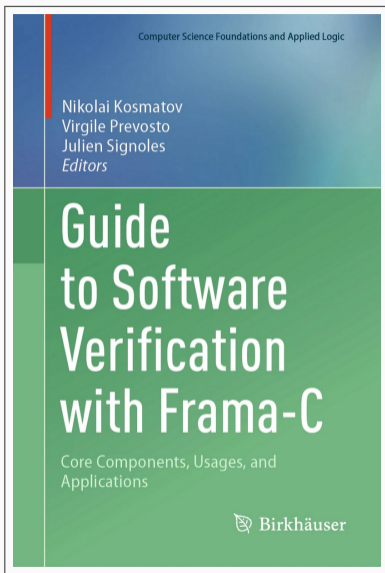
June 13<sup>th</sup>, 2024 @ Frama-C Days

<sup>1</sup> Thales Research & Technology

<sup>2</sup> Université Paris-Saclay, CEA, List, Software Safety and Security Lab



THALES



## Target Readers

- > No formal methods background required
- > Undergraduate CS students
- > C software developers
- > V&V engineers
- > Researchers and teachers in formal methods

## Main Ingredients

- > Present the grounding blocks of the platform
- > Show some advanced and experimental features
- > Describe industrial usages

- > Gentle, example-based **introduction to software specification and verification**
- > Wide panorama of **state-of-the-art** specification and analysis **techniques**
- > Step-by-step **guide to develop your own**, tailor-made analysis on top of the platform
- > Inspiring **success stories** of Frama-C deployment on industrial code
- > **More than 15 years of R&D** on analysis and verification of C code

- > Feb. 2021: initial contacts with Springer
- > Jul. 2021: administrative details completed, CfChap sent
- > Dec. 2021: first chapter submitted
- > Jul. 2023: last chapter submitted
- > Dec. 2023: last revision submitted
- > Jan. 2024: final version sent to Springer
- > Apr. 2024: authors' proofs sent by Springer
- > Jul. 2024: book expected to be printed

- > 726 pages
- > 16 chapters
- > 39 authors
- > 31 reviewers
- > 10 countries
- > Springer's editorial team

**Part I** Core Components and Analyzers

**Part II** Advanced Usages and Analyses

**Part III** Case Studies and Industrial Applications

## Part I Core Components and Analyzers

**Chap.1** Formally Expressing what a Program Should Do: The ACSL Language

**Chap.2** The Heart of Frama-C: The Frama-C Kernel

**Chap.3** Abstract Interpretation with the Eva Plug-in

**Chap.4** Formally Verifying That a Program Does What It Should: The WP Plug-in

**Chap.5** Runtime Annotation Checking with Frama-C: The E-ACSL Plug-in (today 11:30)

**Chap.6** Test Generation with PathCrawler

## Part II Advanced Usages and Analyses

## Part III Case Studies and Industrial Applications



## Part I Core Components and Analyzers

## Part II Advanced Usages and Analyses

Chap. 7 The Art of Developing Frama-C Plug-ins

Chap. 8 Tools for Program Understanding (today 11:00)

Chap. 9 Combining Analyses within Frama-C (today 14:00)

Chap.10 Specification and Verification of High-Level Properties (today 15:50)

Chap.11 Advanced Memory and Shape Analysis (tomorrow 11:00)

Chap.12 Analysis of Embedded Numerical Programs in the presence of Numerical Filters  
(tomorrow 13:30)

## Part III Case Studies and Industrial Applications

**Part I** Core Components and Analyzers

**Part II** Advanced Usages and Analyses

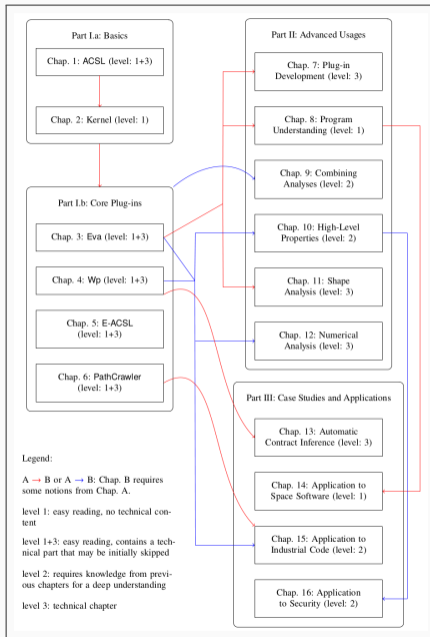
**Part III** Case Studies and Industrial Applications

**Chap.13** An Exercise in Mind Reading: Automatic Contract Inference for Frama-C  
(tomorrow 10:30)

**Chap.14** Exploring Frama-C Resources through Verification of a Space Software

**Chap.15** Ten Years of Industrial Experiments with Frama-C at Mitsubishi Electric R&D Centre Europe (today 16:50)

**Chap.16** Proof of Security Properties: Application to JavaCard Virtual Machine (today 16:20)



We warmly **thank** all people who contributed to Frama-C and the book, notably:

- > Jacques Raguideau & Fabrice Derepas who encouraged this project
- > Benjamin Monate & Pascal Cuoq who initiated the development
- > all computer scientists, including permanent researchers and engineers, postdocs, PhD students, interns, who contributed to the design, development and evaluation
- > ANR, Horizon Europe, BPIFrance, other programs and agencies who supported the development
- > the authors
- > the reviewers
- > Springer editorial team

- > Publication announced for **July 30, 2024**
- > Can be **pre-ordered** on Springer website

<https://link.springer.com/book/9783031556074>

