



**River Publishers**

# **River Publishers Book Catalogue**

Series in Software Engineering

River Publishers Series in Software Engineering

## Requirements Engineering for Safety-Critical Systems

**Editors:**

Luiz Eduardo G. Martins, Federal University of São Paulo, Brazil

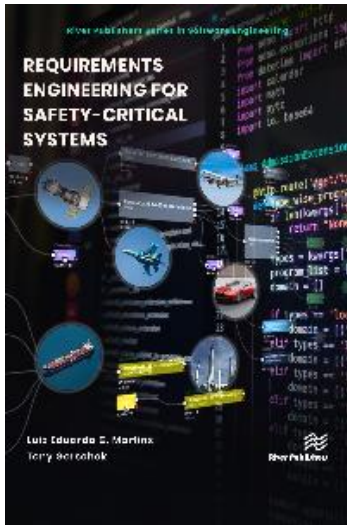
Tony Gorschek, Blekinge Institute of Technology, Sweden

**ISBN:** 9788770224277

**e-ISBN:** 9788770224260

**Available From:** October 2021

**Price:** € 95.00



**Description:**

Safety-Critical Systems (SCS) are increasingly present in people's daily activities. In the means of transport, in medical treatments, in industrial processes, in the control of air, land, maritime traffic, and many other situations, we use and depend on SCS. The requirements engineering of any system is crucial for the proper development of the same, and it becomes even more relevant for the development of SCS. Requirements Engineering is a discipline that focuses on the development of techniques, methods, processes, and tools that assist in the design of software and systems, covering the activities of elicitation, analysis, modeling and specification, validation, and management of requirements. The complete specification of system requirements establishes the basis for its architectural design. It offers a description of the functional and quality aspects that should guide the implementation and system evolution. In this book, we discuss essential elements of requirements engineering applied to SCS, such as the relationship between safety/hazard analysis and requirements specification, a balance between conservative and agile methodologies during SCS development, the role of requirements engineering in safety cases, and requirements engineering maturity model for SCS. This book provides relevant insights for professionals, students, and researchers interested in improving the quality of the SCS development process, making system requirements a solid foundation for improving the safety and security of future systems.

**Keywords:** Requirements engineering, safety-critical systems, safety and security, safety analysis, hazard analysis, safety cases.

River Publishers Series in Software Engineering

## Foundations of Probabilistic Logic Programming Languages, Semantics, Inference and Learning

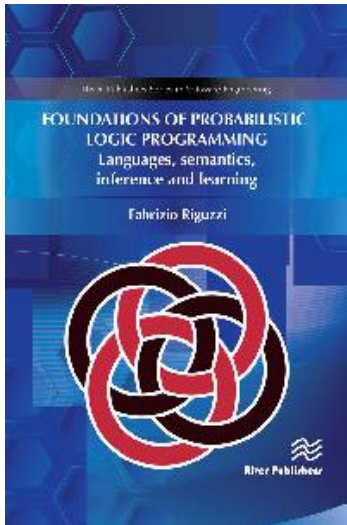
**Author:** Fabrizio Riguzzi, University of Ferrara, Italy

**ISBN:** 9788770220187

**e-ISBN:** 9788770220170

**Available From:** September 2018

**Price:** € 95.00



### Description:

Probabilistic Logic Programming extends Logic Programming by enabling the representation of uncertain information. Probabilistic Logic Programming is at the intersection of two wider research fields: the integration of logic and probability and Probabilistic Programming.

Logic enables the representation of complex relations among entities while probability theory is useful for model uncertainty over attributes and relations. Combining the two is a very active field of study. Probabilistic Programming extends programming languages with probabilistic primitives that can be used to write complex probabilistic models. Algorithms for the inference and learning tasks are then provided automatically by the system.

Probabilistic Logic programming is at the same time a logic language, with its knowledge representation capabilities, and a Turing complete language, with its computation capabilities, thus providing the best of both worlds.

Since its birth, the field of Probabilistic Logic Programming has seen a steady increase of activity, with many proposals for languages and algorithms for inference and learning. *Foundations of Probabilistic Logic Programming* aims at providing an overview of the field with a special emphasis on languages under the Distribution Semantics, one of the most influential approaches. The book presents the main ideas for semantics, inference, and learning and highlights connections between the methods.

Many examples of the book include a link to a page of the web application <http://cplint.eu> where the code can be run online.

**Keywords:** Probabilistic logic programming, statistical relational learning, statistical relational artificial intelligence, distribution semantics, graphical models, artificial intelligence, machine learning

River Publishers Series in Software Engineering

## Cybernetics in C++

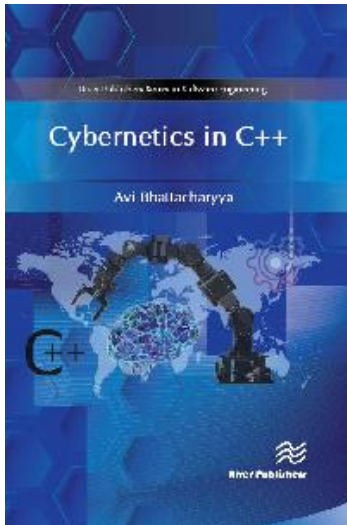
**Author:** Avi Bhattacharyya, Shomiron Institute, UK

**ISBN:** 9788793609457

**e-ISBN:** 9788793609440

**Available From:** July 2018

**Price:** € 85.00



### Description:

C++ is a powerful, much sought after programming language, but can be daunting to work with, even for engineering professionals.

Why is this book so useful? Have you ever wondered:

- How do keywords like static and virtual change their meanings according to context?
- What are the similarities and differences between Pointers and References, Pointers and Arrays, Constructors and Copy Constructors, Nested and Local Inner Classes?
- Why is Multiple Interface Inheritance seen to be beautiful but Multiple Implementation Inheritance considered evil?
- When is Polymorphism Static or Dynamic, Bounded or Unbounded?

Answers on these questions, and much more, are explained in this book, *Cybernetics in C++*. What makes this text so different and appealing in comparison to existing books on the market?

- The Bulleted style, as opposed to Prose, produces results much faster, both in learning and reference
- Rules of Thumb, and further expert Tips are given throughout in how to optimise your code
- The Prospective Evils sections tell you what to avoid
- The thorough coverage ensures you will be trained to expert level in each of Imperative, Procedural, Memory & Resource Management, Object Oriented and Generic Programming

*Cybernetics in C++* combines a theoretical overview and practical approach in one book, which should prove to be a useful reference for computer scientists, software programmers, engineers and students in this and related field.

**Keywords:** C++, Computer Programming, Software Languages, Object Oriented Programming, Imperative Programming, Procedural Programming, Generic Programming