Software Engineering using Formal Methods

Lecture Notes by Thomas Schulz
Last Update: February 9, 2012 - 16:04

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Contents

Disclaimer II

Motivation III

1 Modeling & Model Checking with PROMELA & SPIN 1

1.1 PROMELA Introduction . . . 1
1.2 Verifying with SPIN . . . 2
1.3 Modeling Concurrency . . . 3
1.4 Introduction to PROMELA/SPIN . . . 4
1.5 Modeling Distribution . . . 5
1.6 Propositional Logic & Temporal Logic (1) . . . 6
1.7 Temporal Logic (2) . . . 7
1.8 Channels & Linear Temporal Logic . . . 8
1.9 Temporal Model Checking with SPIN . . . 9

2 Modeling & Verification with JML & KEY 10

2.1 First-Order Logic (Syntax and Semantics) . . . 10
2.2 First-Order Logic – Calculus . . . 11
2.3 JML (1) . . . 12
2.4 JML (2) . . . 13
2.5 Dynamic Logic 1 . . . 14
2.6 Dynamic Logic Calculus . . . 15
2.7 Proof-Obligations . . . 16
2.8 Loop Invariants . . . 17

Disclaimer

This document is a summary of the lecture “Software Engineering using Formal Methods”. It was created for personal study and exam preaparation.

The author do not warrant or assume any legal responsibility for the accuracy, completeness, or usefulness of any information described in this document.

Motivation

1 Modeling & Model Checking with PROMELA & SPIN

1.1 PROMELA Introduction

1.2 Verifying with SPIN

1.3 Modeling Concurrency

1.4 Introduction to PROMELA/SPIN

1.5 Modeling Distribution

1.6 Propositional Logic & Temporal Logic (1)

1.7 Temporal Logic (2)

1.8 Channels & Linear Temporal Logic

1.9 Temporal Model Checking with SPIN

2 Modeling & Verification with JML & KeY

2.1 First-Order Logic (Syntax and Semantics)

2.2 First-Order Logic – Calculus

2.3 JML (1)

2.4 JML (2)

2.5 Dynamic Logic 1

2.6 Dynamic Logic Calculus

2.7 Proof-Obligations

2.8 Loop Invariants