Computing Handbook 2

Computer Science and Software Engineering

Part 8 – Programming Languages
01. Imperative Language Paradigm
1. Introduction
2. Data Bindings: Variables, Type, Scope, and Lifetime (Topics 5)
3. Control Structures (Topics 5)
4. Best Practices (Topics 5)
5. Research Issues and Summary
02. Object-Oriented Language Paradigm
1. Introduction
2. Underlying Principles
3. Impact on Practice (Java | C++ | C# | Objective-C)
4. Language Implementation Issues
5. Summary
03. Logic Programming and Constraint Logic Programming
1. Introduction
2. Introductory Example
3. Features of Logic Programming Languages
4. Historical Remarks
5. Resolution and Unification (Topics 3)
6. Procedural Interpretation: Examples
7. Impure Features
8. Constraint Logic Programming
9. Recent Developments in CLP (2002) (Topics 7)
10. Applications
11. Theoretical Foundations
12. Meta-level Interpretation
13. Implementation (Topics 4)
14. Research Issues (Topics 4)
15. Conclusion
04. Multi-paradigm Languages
1. Introduction
2. Principles of Programming Paradigms (Topics 3)
3. Multi-paradigm Language in Practice (Topics 3)
4. Research Issues
5. Summary
05. Scripting Languages
1. Introduction
2. History (Topics 5)
3. Principles (Topics 4)
4. Conclusions
06. Compilers and Interpreters
1. Introduction
2. Underlying Principles (Topics 4)
3. Best Practices (Topics 9)
4. Incremental Compilation
5. Just-in-Time Compilation (Topics 2)
6. Compilation for Parallel Execution
7. Research Issues and Summary
07. Programming Language Semantics
1. Introduction
2. Underlying Principles of Semantics Methods (Topics 4)
3. Semantical Principles of Programming Languages (Topics 7)
4. Practical Impact
5. Research Issues
08. Type Systems
1. Introduction (Topics 2)
2. Formal Type Systems (Topics 10)
3. Reasoning about Type Systems (Topics 4)
4. Type Inference (Topics 2)
5. Conclusion
09. Formal Methods
1. Introduction

2. Underlying Principles (Formal Methods)
3. Best Practices (Topics 3)
4. Case Study (Topics 8)
5. Technology Transfer and Research Issues

Part 9 – Discipline of Software Engineering

01. Discipline of Software Engineering : An Overview

1. Origins of Software Engineering (Topics 8)
2. Engineering and Software Engineering (Topics 4)
3. Software Engineering Research
4. Elements of Software Engineering
5. What is Used in Practice
6. Where Does One Need Software Engineering
7. Summary

02. Professionalism and Certification

1. What is a Profession
2. Engineering as a Profession
3. Emergence of Software Engineering
4. Software Engineering as a Profession (Topics 3)
5. Professional Qualifications for Software Engineering: Licensure Versus Certification (Topics 2)
6. Conclusion

03. Software Engineering Code of Ethics and Professional Practice

1. Introduction (Brief History and Purpose)
2. Code (Topics 5)
3. Foundations and Functions of the Code (Topics 3)
4. Research (Topics 3)
5. Summary

04. Software Business and Economics

1. Introduction
2. Foundations (Topics 17)
3. Practice
4. Business Value (Outlook: Where is the Software Business Heading)

05. Open Source and Governance: COTS

1. Introduction
2. Underlying Principles (Topics 5)
3. Best Practices (Topics 2)
4. Open Issues (What Constitutes Distribution in the Cloud Computing Environment)
5. Summary

Part 10 – Software Quality and Measurement

01. Evidence –Informed Software Engineering and the Systematic Literature

1. Introduction (Topics 2)
2. Performing a Systematic Literature Review (Topics 3)
3. What SLRs Do We Have (Topics 3)
4. What Do SLRs Tell Us (Topics 2)
5. Conclusion

02. Empirical Software Engineering

1. Introduction and History (Topics 2)
2. Scope of Empirical Knowledge (Topics 2)
3. Four Forms of Primary Study (Topics 4)
4. Secondary Studies
5. Organizing Empirical Studies (Topics 2)
6. Conclusions

03. Software Quality and Model-Based Process Improvement

1. Introduction
2. Underlying Principles (Topics 3)
3. Best Practices (Topics 3)
4. Research Issues (Topics 3)
5. Summary

04. Software Metrics and Measurements

1. Introduction
2. Principles
3. Best Practices (Topics 7)
4. Conclusions and Observations

Part 11 – Software Development Management: Processes and Paradigms
01. Software Development: Management and Business Concepts
1. High-Level Process Concepts (Topics 2)
2. Innovation and Design Strategy (Topics 4)
3. Architecture Strategy (Topics 2)
4. Team Management (Topics 2)
5. Project Management (Topics 3)
6. Quality Assurance (Topics 3)
7. Results from Project Surveys and Conclusions (Topics 3)
02. Project Personnel and Organization
1. Introduction
2. Determining the Types of Personnel Required for Your Project (Topics 7)
3. Organizing Project Personnel (Topics 6)
4. What Makes a Team (Topics 7)
5. Challenges and Research Directions (Topics 4)
6. Summary and Conclusion
03. Project and Process Control
1. Introduction
2. Project Control (Topics 6)
3. Process Control (Topics 4)
4. Summary and Conclusions
04. Agile
1. Introduction
2. Underlying Principles
3. Best Practices (Topics 9)
4. Research Areas (Topics 7)
5. Summary
05. Service-Oriented Development
1. Introduction (Topics 3)
2. Underlying Principles (Topics 2)
3. Best Practices (Topics 4)
4. Research Issues
5. Summary
06. Software Product Lines
1. Introduction
2. About Software Product Lines (Topics 3)
3. Research Issues (Topics 2)
4. Summary and Conclusions

Part 12 – Software Modeling, Analysis, and Design
01. Requirements Elicitation
1. Introduction
2. Context of Relic and the Vocabulary for the Rest of This Chapter
3. RE and Relic as Part of RE
4. Elicitation and Relic as Human Activities
5. Relic for Each Job of the RA (Topics 4)
6. Specific Techniques for Relic (Topics 9)
7. What Can Go Wrong during Relic
8. Myths about Relic
9. Why Invention and Creativity Are Necessary for Relic
10. Empirical Studies of Relic Techniques
11. Conclusions
02. Specification
1. Underlying Principles
2. Best Practices (Topics 3)
3. Research Issues
4. Summary
03. Software Model Checking
1. Introduction
2. Model Checking
3. CEGAR-Based Software Model Checking (Topics 5)
4. Bounded Model Checking of Software (Topics 3)
5. Tools Based on Traditional Model Checking
6. Conclusion
04. Software Design Strategies
1. Introduction

2. Underlying Principles (Topics 4)
3. Best Practices (Topics 3)
4. Research Issues (Topics 3)
5. Summary
05. Software Architecture
1. Underlying Principles (Topics 2)
2. Architecture Modeling
3. Architecture Design Methods (Topics 5)
4. Architecture Evaluation
5. Research Issues (Topics 2)
6. Summary
06. Human-Computer Interfaces for Speech Applications
1. Speech Interface Design (Topics 2)
2. Research Issues and Future Development (Topics 3)
07. Software Assurance
1. Introduction
2. What is Software Assurance (Changing the Assurance Focus to Security)
3. Why Does Software Assurance Matter
4. Software Assurance Principles (Topics 2)
5. Software Assurance Process Models and Practices (Topics 3)
6. Research Framework (Building Assured Systems Framework)
7. Summary
8. Master of Software Assurance Knowledge Areas Mapped to Maturity Levels in the Building Assured Systems