

TABLE OF CONTENTS

Chapter 1: Foundations of Performance Engineering	01-11
1.1 Definition and Scope of Performance Engineering	
1.2 Importance of Non-Functional Requirements (NFRs)	
1.3 Key Metrics: Throughput, Latency, and Scalability	
1.4 Performance vs Scalability vs Reliability	
1.5 Role of Performance Engineering in SDLC	
Chapter 2: Performance Engineering Manifesto	12-23
2.1 Principles of Proactive Performance Design	
2.2 Shift-Left Performance Testing	
2.3 Continuous Performance Validation	
2.4 Engineering Culture and Performance Mindset	
2.5 Case Studies from Large-Scale Systems	
Chapter 3: System Architecture & Performance	24-35
3.1 Monolithic vs Microservices Architecture	
3.2 Fundamentals of Distributed Systems	
3.3 CAP Theorem and Trade-offs	
3.4 Cloud-Native Architectures	
3.5 Architectural Impact on Performance	
Chapter 4: Performance Metrics & Modeling	36-48
4.1 Key Performance Indicators (KPIs)	
4.2 Workload Modeling Techniques	
4.3 Little's Law and Applications	
4.4 Basics of Queueing Theory	
4.5 Capacity Planning Strategies	
4.6 Service Level Agreements (SLA), Objectives (SLO), and Indicators (SLI)	
Chapter 5: Performance Testing Strategies	49-60
5.1 Types of Testing: Load, Stress, Spike, and Soak	
5.2 Test Planning and Scenario Design	
5.3 Test Data Management	
5.4 Performance Testing Tools Overview (JMeter, LoadRunner)	
5.5 Test Environment Configuration	
Chapter 6: Profiling & Bottleneck Analysis	61-72
6.1 CPU, Memory, Disk, and Network Profiling	
6.2 Code-Level Performance Analysis	
6.3 Thread and Heap Analysis	

6.4 Root Cause Analysis Techniques	
6.5 Common Performance Anti-Patterns	
Chapter 7: Database & Data Systems Performance	73-87
7.1 Query Optimization Techniques	
7.2 Indexing Strategies	
7.3 Data Partitioning and Sharding	
7.4 Caching Mechanisms	
7.5 Performance Tuning in SQL and NoSQL Systems	
Chapter 8: Cloud & Distributed Performance Engineering	88-107
8.1 Performance in Cloud Environments	
8.2 Auto-Scaling Strategies	
8.3 Container Performance (Docker and Kubernetes)	
8.4 Observability: Logs, Metrics, and Traces	
8.5 Cost vs Performance Trade-offs	
Chapter 9: Performance Optimization Techniques	108-130
9.1 Algorithmic Optimization	
9.2 Caching Strategies	
9.3 Load Balancing Techniques	
9.4 Asynchronous Processing	
9.5 Parallel and Distributed Computing	
9.6 Performance Tuning Lifecycle	
Chapter 10: DevOps, SRE & Future Trends	131-154
10.1 CI/CD Integration with Performance Engineering	
10.2 Introduction to Site Reliability Engineering (SRE)	
10.3 Chaos Engineering Principles	
10.4 AIOps and Self-Healing Systems	
10.5 Green Computing and Sustainable Performance	
Chapter 11: Performance Engineering in Modern Programming Paradigms	155-178
11.1 Performance Implications of Programming Languages (Java, Python, Go, Rust)	
11.2 Memory Management and Garbage Collection Impact	
11.3 Functional Programming vs Object-Oriented Performance	
11.4 Concurrency Models: Threads, Async, and Event-Driven Architectures	
11.5 Runtime Optimization Techniques and JIT Compilation	
Chapter 12: Real-Time Systems & High-Frequency Performance Engineering	179-195
12.1 Characteristics of Real-Time Systems	
12.2 Hard Real-Time vs Soft Real-Time Constraints	
12.3 Scheduling Algorithms and Deadline Management	

- 12.4 High-Frequency Transaction Systems (FinTech, Trading Systems)
- 12.5 Latency Optimization in Real-Time Environments

Chapter 13: Security-Aware Performance Engineering

196-217

- 13.1 Impact of Security Layers on System Performance
- 13.2 Encryption/Decryption Overheads and Optimization
- 13.3 Secure API Design and Performance Trade-offs
- 13.4 Authentication and Authorization Latency
- 13.5 Performance in Zero-Trust Architectures