



Cloud Native Architecture and Design

A Handbook for Modern Day
Architecture and Design with
Enterprise-Grade Examples

—
Shivakumar R Goniwada

Apress®

Cloud Native Architecture and Design

**A Handbook for Modern Day
Architecture and Design
with Enterprise-Grade Examples**

Shivakumar R Goniwada

Apress®

Cloud Native Architecture and Design: A Handbook for Modern Day Architecture and Design with Enterprise-Grade Examples

Shivakumar R Goniwada
Bangalore, India

ISBN-13 (pbk): 978-1-4842-7225-1
<https://doi.org/10.1007/978-1-4842-7226-8>

ISBN-13 (electronic): 978-1-4842-7226-8

Copyright © 2022 by Shivakumar R Goniwada

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

Trademarked names, logos, and images may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, logo, or image we use the names, logos, and images only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Managing Director, Apress Media LLC: Welmoed Spahr
Acquisitions Editor: Celestin Suresh John
Development Editor: Mark Powers
Coordinating Editor: Shrikant Vishwakarma

Cover designed by eStudioCalamar

Cover image designed by Pexels

Distributed to the book trade worldwide by Springer Science+Business Media LLC, 1 New York Plaza, Suite 4600, New York, NY 10004. Phone 1-800-SPRINGER, fax (201) 348-4505, e-mail orders-ny@springer-sbm.com, or visit www.springeronline.com. Apress Media, LLC is a California LLC and the sole member (owner) is Springer Science + Business Media Finance Inc (SSBM Finance Inc). SSBM Finance Inc is a Delaware corporation.

For information on translations, please e-mail booktranslations@springernature.com; for reprint, paperback, or audio rights, please e-mail bookpermissions@springernature.com, or visit www.apress.com/rights-permissions.

Apress titles may be purchased in bulk for academic, corporate, or promotional use. eBook versions and licenses are also available for most titles. For more information, reference our Print and eBook Bulk Sales web page at www.apress.com/bulk-sales.

Any source code or other supplementary material referenced by the author in this book is available to readers on GitHub via the book's product page, located at www.apress.com/978-1-4842-7225-1. For more detailed information, please visit www.apress.com/source-code.

Printed on acid-free paper

*This book is dedicated to all the unsung heroes and
frontline workers continuously fighting the COVID-19
battle to save humanity and the world.*

Table of Contents

About the Author	xxvii
About the Technical Reviewer	xxix
Acknowledgments	xxx
Introduction	xxxiii
Part I: The Cloud Native Journey, Principles, and Patterns	1
Chapter 1: Introduction to Cloud Native Architecture.....	3
Introduction to Cloud Native.....	4
Cloud Adoption Across Industries	5
Reducing Costs	5
Adopting the Cloud Native Mindset.....	5
What Is Cloud Native?.....	5
Cloud Native Maturity Model.....	7
Cloud Enablement Wave	8
Cloud Native Transformation Wave	9
Scalability and Flexibility Advantage	11
Cloud Native Culture and Innovation Wave	11
Elements of Cloud Native Computing.....	13
Microservices Architecture.....	14
Serverless Architecture	14
Event-Driven Architecture	15
Cloud Computing	15
Containers	16
Agile Development.....	16
DevSecOps	17

TABLE OF CONTENTS

- How Is Cloud Native Different Than Cloud-Enabled? 17
- Cloud Native Journey 17
 - Start with Lift and Shift 18
 - Re-engineer Migration..... 19
- Benefits of Cloud Native..... 19
- Cloud Native Organization and Culture 20
- How Is Cloud Native Architecture Embraced Across Industries? 22
 - Migrate 23
 - Accelerate 24
 - Scale and Innovate 24
- What Is a Software Architect's Role in Cloud Native? 25
- Summary..... 26
- Chapter 2: Cloud Native Services 27**
 - Evolution of Infrastructure Services..... 27
 - Mainframe Services 29
 - Minicomputer Services..... 30
 - Personal Computing Service 30
 - Client-Server Service 30
 - Enterprise Computing Service..... 31
 - Cloud and Mobile Computing Services..... 31
 - IT Infrastructure Laws and Prediction..... 32
 - Moore's Law 32
 - The Laws of Mass Digital Storage 33
 - Metcalf's Law 33
 - Communication Cost and Internet..... 33
 - Evolution of Servers..... 34
 - Bare-Metal Servers 34
 - Virtual Machine Revolution..... 34
 - Container Revolution 37

Understanding Cloud Services	40
Infrastructure as a Service	40
Platform as a Service	42
Software as a Service	45
Cloud Computing Deployment Models	50
Public Cloud.....	51
Private Cloud or On-Premises Cloud.....	51
Community Cloud	52
Hybrid	52
Cloud Services	52
Summary.....	54
Chapter 3: Cloud Native Architecture Principles	55
What Are Architecture Principles?.....	56
Cloud Native Design Principles	57
API First Principle	57
Monolithic Architecture Principle	59
Polyolithic Architecture Principle.....	60
Polyglot Persistence Principle.....	61
Modeled with Business Domain Principle	62
Consumer First Principle	63
Decentralize Everything Principle.....	64
Culture of Automation Principle.....	65
Always Be Architecting Principle.....	66
Interoperability Principle	66
Digital Decoupling Principle	67
Single Source of Truth Principle	69
Evolutionary Design Principle.....	69

TABLE OF CONTENTS

- Cloud Native Runtime Principles..... 70
 - Isolate Failure Principle (IFP)..... 70
 - Deploy Independently Principle 72
 - Be Smart with State Principle 72
 - Location-Independent Principle..... 73
 - Design for Failure Principle 74
- Security Principles 75
 - Defense in Depth Principle 75
 - Security by Design Principle..... 76
- Software Engineering Principle..... 79
 - Products Not Projects Principle..... 79
 - Shift-Left Principle..... 80
- Container Principles..... 81
 - Single Concern Principle 82
 - High Observability Principle 83
 - Lifecycle Conformance Principle 84
 - Image Immutability Principle..... 86
 - Process Disposability Principle (PDP)..... 87
 - Self-Containment Principle..... 90
 - Runtime Confinement Principle..... 91
- Principles of Orthogonal..... 92
 - Cohesion..... 93
 - Coupling 98
- Software Quality Principles..... 105
 - KISS Principle 106
 - Don't Repeat Yourself 107
 - Isolate..... 110
 - Separation of Concern..... 111

Use Layering.....	112
Information Hiding.....	114
You Aren't Gonna Need It.....	115
SOLID Design Principles.....	117
Single Responsibility Principle	117
Open-Closed Principle	119
Liskov Substitution Principle	121
Interface Segregation Principle	123
Dependency Inversion Principle	123
Summary	124
Chapter 4: Cloud Native Architecture and Design Patterns	127
Evolution of Design Patterns.....	128
What Are Software Patterns?.....	130
Architecture Style, Architecture Pattern, and Design Pattern	130
Anti-pattern.....	131
Cloud Native Data Management Pattern for Microservices	131
Event Sourcing Pattern.....	132
Command and Query Responsibility Segregation Pattern.....	135
Data Partitioning Pattern	139
Data Replication	146
Cloud Native API Management Patterns for Microservices.....	152
Idempotent Service Operation.....	153
Optimistic Concurrency Control in API.....	154
Circuit Breaker.....	157
Service Discovery.....	159
Service Versioning.....	162
Cloud Native Event-Driven Patterns for Microservices	165
Asynchronous Nonblocking I/O.....	165
Stream Processing	168

TABLE OF CONTENTS

- Cloud Native Design Pattern for Microservices 170
 - Mediator 170
 - Orchestration 171
 - Strangler Pattern 172
 - Bulkhead Pattern 173
 - Anti-corruption Pattern 176
- Cloud Native Runtime Pattern for Microservices 177
 - Fail Fast 178
 - Retry 179
 - Sidecar 180
 - Init Containers 181
 - Saga Pattern 182
- Summary 187
- Part II: Elements of Cloud Native Architecture and Design..... 189**
- Chapter 5: Microservices Architecture and Design 191**
 - Evolution of Microservices 192
 - What Is a Microservices Architecture? 192
 - Characteristics of Microservices 193
 - Organized Around Business Capabilities 193
 - Autonomous 197
 - Smart Endpoints and Dumb Pipes 198
 - Resilience in Microservices 201
 - Elasticity in Microservices 205
 - Distributed State 207
 - Independently Deployable 210
 - Decentralization 211
 - Automation 212
 - Containerization 213
 - Design for Failure 214

Living Continuous Design	216
Self-Healing.....	217
Hexagonal Architecture.....	220
Enterprise Microservices Examples.....	223
Case Study: Trade Finance.....	223
Case Study: Collateral Management.....	227
Microservices and User Interface: Micro Front End	230
Routing	232
Composition.....	232
Communication	232
Pros and Cons of Micro Front Ends	233
Microservice Architecture in Artificial Intelligence	233
AI Subcategories	234
Summary.....	240
Chapter 6: Event-Driven Architecture.....	241
Evolution of Event-Driven Architecture	242
Tightly Coupled World to Loosely Coupled World.....	242
Message Broker World to Event World	243
Event	244
Business Events	245
Technical Events.....	245
Processing an Event	246
Event Handling in Domain Context	247
Event Governance.....	247
What Is Event-Driven Architecture?	248
How Does Event-Driven Architecture Work?	248
Event-Driven Topologies.....	250
Mediator Topology	250
Broker Topology	252
Characteristics of Event-Driven Architecture	253

TABLE OF CONTENTS

- Event-Driven Messaging Models 254
 - Event Messaging 254
 - Event Streaming 254
- Event Processing Styles..... 255
 - Simple Event Processing 255
 - Event Stream Processing 255
 - Complex Event Processing 256
- Event-Driven Architecture Maturity Model..... 257
- Decoupling Use Case by Using Event-Driven Architecture..... 259
 - Make Data Accessible 261
- Real-Time Interactivity 265
- How to Use Existing Message Queues with Event Streams?..... 266
- Transaction Management in Event-Driven Microservices..... 268
 - Two-Phase Commit in Cloud Native Services..... 271
 - Transactions with Events..... 274
- Event-Driven Microservices Interaction..... 277
- Interaction Between Microservices 280
 - Service Mesh..... 281
 - Event Mesh..... 283
- Box- and Port-Style Event-Driven Architecture..... 288
 - Characteristics of Box- and Port-Style Architecture..... 290
- DevOps for Events..... 291
- Event Security 291
 - Field-Level Encryption Consideration..... 292
- Cloud Events 292
- Summary..... 294
- Chapter 7: Serverless Architecture 295**
 - Evolution of Serverless 296
 - What Is Serverless Computing? 297

Essential Components of Serverless.....	299
Serverless and Event-Driven Computing	300
Serverless Design Principles	300
Stateless Functions	301
Push-Based and Event-Driven Pipelines	301
Config: Store Config in the Environment.....	301
Backing Services: Treat Backing Services as Attached Resources	301
Concurrency: Scaling Out via the Process Model	302
Disposability: Maximize Robustness with Quick Startup and Shutdown.....	302
Key Considerations for Serverless Computing.....	302
Why Use Serverless Architecture?	304
Best Practices of Serverless Architecture.....	305
Types of Serverless Architecture	307
Function as a Service	307
Backend as a Service or Mobile Backend as a Service.....	317
Function Deployment.....	319
When to Use Serverless	320
Advantages of Serverless Architecture	321
Reduced Operational Cost	321
Optimized Resource Utilization.....	322
Faster Time to Market	322
Ability to Focus on User Experience	322
Fits with Microservices	322
The Drawbacks of Serverless Architecture.....	322
Standardization	322
Operations Management	323
Tooling Support.....	323
Security	323

TABLE OF CONTENTS

- Long-Term Tasks 323
- Future of Serverless..... 323
- Summary..... 324
- Chapter 8: Cloud Native Data Architecture 325**
- Rethinking Data in a Cloud Native World 326
- Cloud Native Data Persistence Layer 327
- Cloud Native Data Characteristics 328
- How to Select a Data Store 329
- Objects, Files, and Blocks..... 329
- Databases..... 330
- Data Replication..... 344
- Physical Database Replication 344
- Logical Database Replication 345
- Extract, Transfer, and Load 349
- Decoupling Big Data Management from Distributed Data Meshes..... 350
- Step 1: Self-Service Data Infrastructure as a Platform 354
- Step 2: Data as a Product..... 355
- Step 3: Data Infrastructure as a Platform..... 355
- Step 4: Domain-Oriented Decentralized Data Ownership and Architecture 356
- Step 5: Data Governance 356
- Data Processing with Real-Time Streaming for Analytics..... 357
- Lambda Architecture 358
- Kappa Architecture 360
- Microservices in Data Processing with Real-Time Streaming for Analytics..... 360
- Mobile Platform Database..... 361
- Intelligent Data Governance and Compliance in the Cloud Native World..... 363
- Why Data Governance? 363
- What Is Data Governance? 364
- Governance Framework 365
- Summary..... 368

Chapter 9: Designing for “-ilities”	371
Why Do You Need “-ilities”?	372
Partial List of “-ilities”	373
Designing for Security	373
Defense in Depth	374
The CIA Triad	374
Policy as Code	375
Zero-Trust Security	376
Decentralized Identity	377
Validating Input	377
Design for Threats	377
Naive Password Complexity Requirements	378
Compliance as Code	378
Shift-Left Security	378
Single Pane of Glass for Audit	379
Homomorphic Encryption	379
Fail Securely	379
Secure APIs	380
Designing for Elasticity	380
Designing for Resilience	381
Designing for Sustainability	382
The JEVONS Paradox in Cloud Native	382
Software Engineering	385
Sustainability Assessment	386
Designing for Failure	387
Infrastructure	388
Communication	388
Dependencies	388
Internal	388

TABLE OF CONTENTS

- Designing for Reliability 389
 - Pareto Chart..... 391
- Designing for High Availability 392
 - Active-Active Deployments..... 394
 - Active-Passive Deployments 394
- Designing for the Customer 395
- Designing for Interoperability..... 397
- Designing for Events 399
- Designing for Observability..... 400
- Designing for Portability..... 401
- Designing for Ethics 402
- Designing for Accessibility..... 405
 - Accessibility Guidelines and Standards..... 406
- Designing for Automation..... 407
- Designing for Maintainability 408
- Designing for Usability 408
- Summary..... 409

- Part III: Modernizing Enterprise IT Systems 411**

- Chapter 10: Modernize Monolithic Applications to Cloud Native 413**
- What Is Decoupling? 414
- Technical Debt..... 415
 - How Are Technical Debts Accumulated? 415
 - How Is Technical Debt Impacting Your Enterprise? 416
 - How to Decide on Decoupling?..... 417
- Decoupling 418
 - Decoupling Approach..... 422
 - Decoupling Plan..... 424
 - Decoupling Principles 425
 - Decoupling Business Case 425
 - Decoupling Strategies 426

Domain-Driven Design	427
How Does Domain-Driven Design Manage Complexity?	428
What Is a Domain?.....	429
Goals of Domain-Driven Design.....	429
Domain-Driven Design Model.....	430
Guiding Principles of DDD.....	431
Event Storming	432
Key Roles in an Event Storming Workshop.....	433
Event Storming Exercise	434
Value of Domain-Driven Design.....	447
Summary.....	449
Chapter 11: Enterprise IT Assessment for a Cloud Native Journey	451
Introduction.....	452
Assessment.....	453
What Is an Assessment Used For?	453
Assessment Objectives	454
Assessment Execution Approach and Key Activities	455
Cloud Native Assessment.....	456
When to Consider a Cloud Native Assessment.....	457
Cloud Native Maturity Assessment Model	458
Detailed Architecture Assessment	464
Assessment Usage	464
Architecture Assessment Model	464
Assessment Questions Template.....	466
Automation Maturity Assessment	472
Automation Maturity Assessment Model.....	472
Automation Maturity Assessment Questionnaire Template	473
Summary.....	478

- Chapter 12: “-ilities” Fitness Function 479**
- What Is a Fitness Function?..... 480
- Categories of Fitness Functions..... 481
 - Atomic vs. Holistic..... 481
 - Triggered vs. Continuous 481
 - Static vs. Dynamic..... 482
 - Automated vs. Manual..... 482
 - Temporal..... 482
 - International vs. Emergent 482
 - Domain-Specific 483
 - Design-Time Fitness Function..... 483
 - Runtime Fitness Function..... 483
- Execution of the Fitness Function..... 483
 - Manual Execution 484
 - Automated Execution..... 484
- Fitness Function Identification..... 485
 - Fitness Function: Coupling and Cohesion..... 485
 - Fitness Function: Security 487
 - Fitness Function: Extensibility, Reusability, Adaptability, and Maintainability..... 487
 - Fitness Function: Performance..... 488
 - Fitness Function: Resiliency 488
 - Fitness Function: Scalability..... 488
 - Fitness Function: Observability..... 488
 - Fitness Function: Compliance..... 489
- Fitness Function Metrics..... 489
- Review Function Metrics..... 491
- Summary..... 491

Part IV: Cloud Native Software Engineering	493
Chapter 13: Enterprise Cloud Native Software Engineering	495
Cloud Native and Traditional Application Engineering.....	496
Intelligent Software Engineering.....	497
From Project to Product	499
Organization Transformation	500
Agile Software Development Methodologies	502
Hypothesis-Driven Development	502
Test-Driven Development	506
Behavior-Driven Development.....	510
Feature-Driven Development.....	515
Architecture in the Agile Methodology	519
Waterfall to Agile Transformation.....	520
Summary.....	521
Chapter 14: Enterprise Cloud Native Automation	523
Introduction.....	524
DevOps Today and Tomorrow	525
From DevOps to DevSecOps	527
Driver for Shift-Left Security	528
Automation Principles and Best Practices	529
Site Reliability Engineering	530
DevSecOps.....	531
Continuous Integration	531
Continuous Delivery.....	532
Continuous Deployment.....	533
DataOps.....	533
DataOps Principles	535
DataOps Pipeline	536

TABLE OF CONTENTS

- DevNetOps 538
 - Network Operation and Challenges 538
 - Why You Need DevNetOps? 539
 - Network Reliability Engineering 540
 - DevNetOps Pipeline 541
- DevOps in the Cloud 543
 - AWS Cloud 544
 - Azure Cloud 546
 - Google Cloud 548
- DevOps Transformation 549
- Summary 552
- Chapter 15: AI-Driven Development 555**
 - Introduction 555
 - Unique AI Challenges 557
 - Why AI-Driven Development? 557
 - AI-Driven Principles at a Glance 558
 - Approach to AI 559
 - AI Governance 559
 - AI Framework 559
 - AI Governance Measurement 560
 - Governance Process 560
 - Governance Model 560
 - How to Train AI-Enabled Frameworks? 562
 - AI-Driven Methodology 562
 - AI Use Cases 563
 - Discovery and Piloting 564
 - AI Project Execution 565
 - Deploy and Industrialize 565

AI and ML in DevOps	565
AI and ML in Code Management.....	566
Summary.....	570
Part V: Cloud Native Infrastructure.....	571
Chapter 16: Containerization and Virtualization	573
Introduction.....	574
What Is Cloud Native Infrastructure?.....	576
Cloud Native Environment Characteristics	577
Cloud Virtualization	578
How Does Virtualization Work?	578
Types of Virtualization in the Cloud.....	579
What Applications and Services Are Commonly Virtualized?	580
Cloud Native and Virtual Machines.....	582
Containerization	583
What Is a Container Image?	584
Container Architecture.....	585
Container Principles	587
Container Patterns.....	588
Container Benefits.....	592
Container Adoption Best Practices	593
Containers in an Enterprise	593
Container Orchestration	596
Types of Orchestration Tools.....	597
Kubernetes Features	602
Kubernetes Principles and Patterns	603
Running a Cloud Native Application on the Container and Kubernetes Strategy.....	607
Kubernetes Maturity Model	609
Service Meshes and Kubernetes.....	611
Stateful Workloads on Kubernetes	612

TABLE OF CONTENTS

- Kubernetes Multitenancy..... 613
- Kubernetes Secrets 614
- Kubernetes as a Service..... 615
- Summary..... 617
- Chapter 17: Infrastructure Automation 619**
- What Is Infrastructure Automation? 619
- What Can You Automate? 620
- What Is Infrastructure as Code?..... 621
- IaC in Build Pipeline Automation 622
- Capture Requirements..... 623
- Prepare Automation Code..... 623
- Set Up Infrastructure 623
- Install OS 623
- Set Up Network and Storage 624
- Deploy Services 624
- Define Everything As Code..... 624
- How Do You Select an IaC Tool? 625
- What Coding Language Can You Use? 625
- IaC Example 626
- IaC Tools..... 627
- Terraform..... 627
- Ansible..... 629
- SaltStack 629
- Chef 630
- Puppet 630
- CFEngine 631
- AWS Cloud Formation 632
- IaC Tools Comparison 632
- Summary..... 634

Part VI: Cloud Native Operations.....	635
Chapter 18: Intelligent Operations.....	637
Introduction.....	638
Why Do You Need Intelligent Operations?	639
Elements of Intelligent Operation.....	640
Data-Driven Approach	640
Applied Intelligence.....	641
Cloud Enablement	641
Right Talent and Skill.....	641
Smart Partnership	642
AIOps.....	642
Central Functions	643
Example Use Case of AIOps.....	648
Traditional Operations.....	648
AIOps-Based Operation	649
Capabilities of AIOps.....	649
AIOps Transformation	650
Benefits of AIOps	652
ChatOps.....	652
ChatOps Benefits	653
Types of ChatOps.....	654
ChatOps in Service Support.....	656
ChatOps (Bot) Architecture	656
Industry Example Use Cases	658
Summary.....	659

TABLE OF CONTENTS

- Chapter 19: Observability 661**
 - Introduction..... 662
 - Difference Between Monitoring and Observability..... 663
 - Full-Stack Observability 664
 - Connected Across Capabilities 665
 - One Source of Truth 666
 - Visualization 667
 - Observability and Cloud Native Services 668
 - Observability in Kubernetes 669
 - Observability and DevOps 671
 - Common Use Cases for Observability with AIOps..... 671
 - Guidance to Choose Observation Tools 672
 - Benefits of Observability 673
 - Observability, Monitoring, and Machine Learning Models..... 674
 - Algorithms Help in Observability 674
 - Workflow Steps for ML 675
 - Summary..... 676
- Part VII: Cloud Native Features 677**
- Chapter 20: Cloud Native Trends 679**
 - Cloud Native Trends 680
 - Designing for “-ilities” 680
 - Cloud Native Architecture 680
 - Open Application Model Specification 681
 - Web Assembly 682
 - Data Gateways..... 682
 - HTTP/3 683
 - RSocket and Reactive Streams 683
 - Low Code/No Code 684
 - Actor Model 684

Kubernetes on the Edge	685
GitOps	685
General Trends Across Industry.....	686
5G	686
Digital Twin	689
Quantum Computing.....	691
Extended Reality	693
Edge Computing	694
Summary.....	695
Index.....	697