
Building Distributed Systems And Microservices In Go With

Microservices for the Enterprise
Production-Ready Microservices
Hands-On Microservices with Spring Boot and Spring Cloud
Building Microservices with Go
Building Evolutionary Architectures
Distributed Services with Go
Distributed Systems
Microservices Security in Action
Domain-driven Design
Guide to Reliable Distributed Systems
Release It!
Microservices in Action
Microservices: Up and Running
Your Code as a Crime Scene
Designing Microservices Platforms with NATS
Reactive Systems in Java
Microservices with Azure
Microservices Patterns
Building Microservices
Building Event-Driven Microservices
Practical Microservices Architectural Patterns
Monolith to Microservices
Distributed Systems with Node.js
Distributed Tracing in Practice
Mastering Spring Cloud
Building Microservices
Microservices from Theory to Practice: Creating Applications in IBM Bluemix Using
the Microservices Approach
Understanding Distributed Systems, Second Edition
Kubernetes Native Microservices with Quarkus and MicroProfile
Service Design Patterns
Designing Distributed Systems
The Tao of Microservices
Designing Data-Intensive Applications
SOA Source Book
The Art of Scalability
Mastering Distributed Tracing
Building Microservices with JavaScript
Reactive Messaging Patterns with the Actor Model
Microservice Architecture

Building Distributed Applications in Gin

*Building Distributed
Systems And
Microservices In Go
With*

*Downloaded from
blog.gmercyyu.edu by
guest*

HURLEY WINTERS

Microservices for the Enterprise "O'Reilly Media, Inc."

This book describes the key concepts, principles and implementation options for creating high-assurance cloud computing solutions. The guide starts with a broad technical overview and basic introduction to cloud computing, looking at the overall architecture of the cloud, client systems, the modern Internet and cloud computing data centers. It then delves into the core challenges of showing how reliability and fault-tolerance can be abstracted, how the resulting questions can be solved, and how the solutions can be leveraged to create a wide range of practical cloud applications. The author's style is practical, and the guide should be readily understandable without any special background. Concrete examples are often drawn from real-world settings to illustrate key insights. Appendices show how the most important reliability models can be formalized, describe the API of the Isis2 platform, and offer more than 80 problems at varying levels of difficulty.

Production-Ready Microservices

Pragmatic Bookshelf

USE THE ACTOR MODEL TO BUILD
SIMPLER SYSTEMS WITH BETTER
PERFORMANCE AND SCALABILITY

Enterprise software development has been much more difficult and failure-prone than it needs to be. Now, veteran software engineer and author Vaughn Vernon offers an easier and more rewarding method to succeeding with

Actor model. Reactive Messaging Patterns with the Actor Model shows how the reactive enterprise approach, Actor model, Scala, and Akka can help you overcome previous limits of performance and scalability, and skillfully address even the most challenging non-functional requirements. Reflecting his own cutting-edge work, Vernon shows architects and developers how to translate the longtime promises of Actor model into practical reality. First, he introduces the tenets of reactive software, and shows how the message-driven Actor model addresses all of them—making it possible to build systems that are more responsive, resilient, and elastic. Next, he presents a practical Scala bootstrap tutorial, a thorough introduction to Akka and Akka Cluster, and a full chapter on maximizing performance and scalability with Scala and Akka. Building on this foundation, you'll learn to apply enterprise application and integration patterns to establish message channels and endpoints; efficiently construct, route, and transform messages; and build robust systems that are simpler and far more successful. Coverage Includes How reactive architecture replaces complexity with simplicity throughout the core, middle, and edges The characteristics of actors and actor systems, and how Akka makes them more powerful Building systems that perform at scale on one or many computing nodes Establishing channel mechanisms, and choosing appropriate channels for each application and integration challenge Constructing messages to clearly convey a sender's intent in communicating with a receiver Implementing a Process Manager for

your Domain-Driven Designs Decoupling a message's source and destination, and integrating appropriate business logic into its router Understanding the transformations a message may experience in applications and integrations Implementing persistent actors using Event Sourcing and reactive views using CQRS Find unique online training on Domain-Driven Design, Scala, Akka, and other software craftsmanship topics using the [for{comprehension}](http://forcomprehension.com) website at forcomprehension.com.

Hands-On Microservices with Spring Boot and Spring Cloud

IBM Redbooks Organizations today often struggle to balance business requirements with ever-increasing volumes of data. Additionally, the demand for leveraging large-scale, real-time data is growing rapidly among the most competitive digital industries. Conventional system architectures may not be up to the task. With this practical guide, you'll learn how to leverage large-scale data usage across the business units in your organization using the principles of event-driven microservices. Author Adam Bellemare takes you through the process of building an event-driven microservice-powered organization. You'll reconsider how data is produced, accessed, and propagated across your organization. Learn powerful yet simple patterns for unlocking the value of this data. Incorporate event-driven design and architectural principles into your own systems. And completely rethink how your organization delivers value by unlocking near-real-time access to data at scale. You'll learn: How to leverage event-driven architectures to deliver exceptional business value The role of microservices in supporting event-driven designs Architectural patterns to ensure success both within and between teams

in your organization Application patterns for developing powerful event-driven microservices Components and tooling required to get your microservice ecosystem off the ground

Building Microservices with Go

"O'Reilly Media, Inc."

Architect enterprise-grade, Microservice-based solutions using Microsoft Azure Service Fabric. About This Book Explore architectural patterns for building modern day Microservice-based systems Learn about Microsoft Service Fabric as a platform to host distributed Microservices Discover multiple options for hosting Microservices on heterogeneous, cross-platform environments Learn to configure Azure Service Fabric clusters for enterprise-grade service deployments Who This Book Is For The book is aimed at IT architects, system administrators, and DevOps engineers who have a basic knowledge of the Microsoft Azure platform and are working on, or are curious about, the concepts of Microservices and Microservice architecture. What You Will Learn Understand the basics of Microservices and how Microsoft Azure fits into the equation Master Azure Service Fabric architecture and services Explore Azure Service Fabric application programming models Comprehensive study of various architecture patterns for building enterprise-grade Microservices Manage and deploy Microservices on Azure Service Fabric An insight into the future of Microservices with containers and serverless computing In Detail Microsoft Azure is rapidly evolving and is widely used as a platform on which you can build Microservices that can be deployed on-premise and on-cloud heterogeneous environments through Microsoft Azure Service Fabric. This book will help you

understand the concepts of Microservice application architecture and build highly maintainable and scalable enterprise-grade applications using the various services in Microsoft Azure Service Fabric. We will begin by understanding the intricacies of the Microservices architecture and its advantages over the monolithic architecture and Service Oriented Architecture (SOA) principles. We will present various scenarios where Microservices should be used and walk you through the architectures of Microservice-based applications. Next, you will take an in-depth look at Microsoft Azure Service Fabric, which is the best-in-class platform for building Microservices. You will explore how to develop and deploy sample applications on Microsoft Azure Service Fabric to gain a thorough understanding of it. Building Microservice-based application is complicated. Therefore, we will take you through several design patterns that solve the various challenges associated with realizing the Microservices architecture in enterprise applications. Each pattern will be clearly illustrated with examples that you can keep referring to when designing applications. Finally, you will be introduced to advanced topics such as Serverless computing and DevOps using Service Fabric, to help you undertake your next venture with confidence. Style and approach This book introduces its readers to the concept of Microservices and Microsoft Azure Service Fabric as a distributed platform to host enterprise-grade Microservices. It then addresses common architectural challenges associated with the Microservice architecture, using proven architectural patterns.

Building Evolutionary Architectures Packt Publishing Ltd

How do you detangle a monolithic system and migrate it to a microservice architecture? How do you do it while maintaining business-as-usual? As a companion to Sam Newman's extremely popular *Building Microservices*, this new book details a proven method for transitioning an existing monolithic system to a microservice architecture. With many illustrative examples, insightful migration patterns, and a bevy of practical advice to transition your monolith enterprise into a microservice operation, this practical guide covers multiple scenarios and strategies for a successful migration, from initial planning all the way through application and database decomposition. You'll learn several tried and tested patterns and techniques that you can use as you migrate your existing architecture. Ideal for organizations looking to transition to microservices, rather than rebuild Helps companies determine whether to migrate, when to migrate, and where to begin Addresses communication, integration, and the migration of legacy systems Discusses multiple migration patterns and where they apply Provides database migration examples, along with synchronization strategies Explores application decomposition, including several architectural refactoring patterns Delves into details of database decomposition, including the impact of breaking referential and transactional integrity, new failure modes, and more

Distributed Services with Go Van Haren

Apply microservices patterns to build resilient and scalable distributed systems Key Features Understand the challenges of building large-scale microservice landscapes Build cloud-native production-ready microservices with this comprehensive guide Discover

how to get the best out of Spring Cloud, Kubernetes, and Istio when used together. Book Description Microservices architecture allows developers to build and maintain applications with ease, and enterprises are rapidly adopting it to build software using Spring Boot as their default framework. With this book, you'll learn how to efficiently build and deploy microservices using Spring Boot. This microservices book will take you through tried and tested approaches to building distributed systems and implementing microservices architecture in your organization. Starting with a set of simple cooperating microservices developed using Spring Boot, you'll learn how you can add functionalities such as persistence, make your microservices reactive, and describe their APIs using Swagger/OpenAPI. As you advance, you'll understand how to add different services from Spring Cloud to your microservice system. The book also demonstrates how to deploy your microservices using Kubernetes and manage them with Istio for improved security and traffic management. Finally, you'll explore centralized log management using the EFK stack and monitor microservices using Prometheus and Grafana. By the end of this book, you'll be able to build microservices that are scalable and robust using Spring Boot and Spring Cloud. What you will learn Build reactive microservices using Spring Boot Develop resilient and scalable microservices using Spring Cloud Use OAuth 2.0/OIDC and Spring Security to protect public APIs Implement Docker to bridge the gap between development, testing, and production Deploy and manage microservices using Kubernetes Apply Istio for improved security, observability, and traffic management Who this book is

for This book is for Java and Spring developers and architects who want to learn how to break up their existing monoliths into microservices and deploy them either on-premises or in the cloud using Kubernetes as a container orchestrator and Istio as a service Mesh. No familiarity with microservices architecture is required to get started with this book.

Distributed Systems Addison-Wesley Professional

Reactive systems and event-driven architecture are becoming indispensable to application design, and companies are taking note. Reactive systems ensure that applications are responsive, resilient, and elastic no matter what failures or errors may be occurring, while event-driven architecture offers a flexible and composable option for distributed systems. This practical book helps Java developers bring these approaches together using Quarkus 2.x, the Kubernetes-native Java framework. Clement Escoffier and Ken Finnigan show you how to take advantage of event-driven and reactive principles to build robust distributed systems, reducing latency and increasing throughput, particularly in microservices and serverless applications. You'll also get a foundation in Quarkus to help you create true Kubernetes-native applications for the cloud. Understand the fundamentals of reactive systems and event-driven architecture Learn how to use Quarkus to build reactive applications Combine Quarkus with Apache Kafka or AMQP to build reactive systems Develop microservices that utilize messages with Quarkus for use in event-driven architectures Learn how to integrate external messaging systems, such as Apache Kafka, with Quarkus Build applications with Quarkus using reactive

systems and reactive programming concepts

Microservices Security in Action Roberto Vitillo

"Forewords by Martin Fowler and Ian Robinson"--From front cover.

Domain-driven Design "O'Reilly Media, Inc."

Have you heard about the tremendous success Amazon and Netflix have had by switching to a microservice architecture? Are you wondering how this can benefit your company? Or are you skeptical about how it might work? If you've answered yes to any of these questions, this practical book will benefit you. You'll learn how to take advantage of the microservice architectural style for building systems, and learn from the experiences of others to adopt and execute this approach most successfully. *Guide to Reliable Distributed Systems* "O'Reilly Media, Inc."

Distributed systems have become more fine-grained in the past 10 years, shifting from code-heavy monolithic applications to smaller, self-contained microservices. But developing these systems brings its own set of headaches. With lots of examples and practical advice, this book takes a holistic view of the topics that system architects and administrators must consider when building, managing, and evolving microservice architectures. Microservice technologies are moving quickly. Author Sam Newman provides you with a firm grounding in the concepts while diving into current solutions for modeling, integrating, testing, deploying, and monitoring your own autonomous services. You'll follow a fictional company throughout the book to learn how building a microservice architecture affects a single domain. Discover how microservices allow you to align your system design with your

organization's goals Learn options for integrating a service with the rest of your system Take an incremental approach when splitting monolithic codebases Deploy individual microservices through continuous integration Examine the complexities of testing and monitoring distributed services Manage security with user-to-service and service-to-service models Understand the challenges of scaling microservice architectures

Release It! Packt Publishing Ltd

Learning to build distributed systems is hard, especially if they are large scale. It's not that there is a lack of information out there. You can find academic papers, engineering blogs, and even books on the subject. The problem is that the available information is spread out all over the place, and if you were to put it on a spectrum from theory to practice, you would find a lot of material at the two ends but not much in the middle. That is why I decided to write a book that brings together the core theoretical and practical concepts of distributed systems so that you don't have to spend hours connecting the dots. This book will guide you through the fundamentals of large-scale distributed systems, with just enough details and external references to dive deeper. This is the guide I wished existed when I first started out, based on my experience building large distributed systems that scale to millions of requests per second and billions of devices. If you are a developer working on the backend of web or mobile applications (or would like to be!), this book is for you. When building distributed applications, you need to be familiar with the network stack, data consistency models, scalability and reliability patterns, observability best practices, and much more. Although you

can build applications without knowing much of that, you will end up spending hours debugging and re-architecting them, learning hard lessons that you could have acquired in a much faster and less painful way. However, if you have several years of experience designing and building highly available and fault-tolerant applications that scale to millions of users, this book might not be for you. As an expert, you are likely looking for depth rather than breadth, and this book focuses more on the latter since it would be impossible to cover the field otherwise. The second edition is a complete rewrite of the previous edition. Every page of the first edition has been reviewed and where appropriate reworked, with new topics covered for the first time.

Microservices in Action Addison-Wesley Professional

The Comprehensive, Proven Approach to IT Scalability—Updated with New Strategies, Technologies, and Case Studies In The Art of Scalability, Second Edition, leading scalability consultants Martin L. Abbott and Michael T. Fisher cover everything you need to know to smoothly scale products and services for any requirement. This extensively revised edition reflects new technologies, strategies, and lessons, as well as new case studies from the authors' pioneering consulting practice, AKF Partners. Writing for technical and nontechnical decision-makers, Abbott and Fisher cover everything that impacts scalability, including architecture, process, people, organization, and technology. Their insights and recommendations reflect more than thirty years of experience at companies ranging from eBay to Visa, and Salesforce.com to Apple. You'll find updated strategies for structuring

organizations to maximize agility and scalability, as well as new insights into the cloud (IaaS/PaaS) transition, NoSQL, DevOps, business metrics, and more. Using this guide's tools and advice, you can systematically clear away obstacles to scalability—and achieve unprecedented IT and business performance. Coverage includes • Why scalability problems start with organizations and people, not technology, and what to do about it • Actionable lessons from real successes and failures • Staffing, structuring, and leading the agile, scalable organization • Scaling processes for hyper-growth environments • Architecting scalability: proprietary models for clarifying needs and making choices—including 15 key success principles • Emerging technologies and challenges: data cost, datacenter planning, cloud evolution, and customer-aligned monitoring • Measuring availability, capacity, load, and performance

Microservices: Up and Running

"O'Reilly Media, Inc."

Build fast, efficient Kubernetes-based Java applications using the Quarkus framework, MicroProfile, and Java standards. In Kubernetes Native Microservices with Quarkus and MicroProfile you'll learn how to: Deploy enterprise Java applications on Kubernetes Develop applications using the Quarkus runtime Compile natively using GraalVM for blazing speed Create efficient microservices applications Take advantage of MicroProfile specifications Popular Java frameworks like Spring were designed long before Kubernetes and the microservices revolution. Kubernetes Native Microservices with Quarkus and MicroProfile introduces next generation tools that have been cloud-native and Kubernetes-aware right from

the beginning. Written by veteran Java developers John Clingan and Ken Finnigan, this book shares expert insight into Quarkus and MicroProfile directly from contributors at Red Hat. You'll learn how to utilize these modern tools to create efficient enterprise Java applications that are easy to deploy, maintain, and expand. About the technology Build microservices efficiently with modern Kubernetes-first tools! Quarkus works naturally with containers and Kubernetes, radically simplifying the development and deployment of microservices. This powerful framework minimizes startup time and memory use, accelerating performance and reducing hosting cost. And because it's Java from the ground up, it integrates seamlessly with your existing JVM codebase. About the book Kubernetes Native Microservices with Quarkus and MicroProfile teaches you to build microservices using containers, Kubernetes, and the Quarkus framework. You'll immediately start developing a deployable application using Quarkus and the MicroProfile APIs. Then, you'll explore the startup and runtime gains Quarkus delivers out of the box and also learn how to supercharge performance by compiling natively using GraalVM. Along the way, you'll see how to integrate a Quarkus application with Spring and pick up pro tips for monitoring and managing your microservices. What's inside Deploy enterprise Java applications on Kubernetes Develop applications using the Quarkus runtime framework Compile natively using GraalVM for blazing speed Take advantage of MicroProfile specifications About the reader For intermediate Java developers comfortable with Java EE, Jakarta EE, or Spring. Some experience with Docker

and Kubernetes required. About the author John Clingan is a senior principal product manager at Red Hat, where he works on enterprise Java standards and Quarkus. Ken Finnigan is a senior principal software engineer at Workday, previously at Red Hat working on Quarkus. Table of Contents PART 1 INTRODUCTION 1 Introduction to Quarkus, MicroProfile, and Kubernetes 2 Your first Quarkus application PART 2 DEVELOPING MICROSERVICES 3 Configuring microservices 4 Database access with Panache 5 Clients for consuming other microservices 6 Application health 7 Resilience strategies 8 Reactive in an imperative world 9 Developing Spring microservices with Quarkus PART 3 OBSERVABILITY, API DEFINITION, AND SECURITY OF MICROSERVICES 10 Capturing metrics 11 Tracing microservices 12 API visualization 13 Securing a microservice *Your Code as a Crime Scene* Packt Publishing Ltd

Microservices architectures offer faster change speeds, better scalability, and cleaner, evolvable system designs. But implementing your first microservices architecture is difficult. How do you make myriad choices, educate your team on all the technical details, and navigate the organization to a successful execution to maximize your chance of success? With this book, authors Ronnie Mitra and Irakli Nadareishvili provide step-by-step guidance for building an effective microservices architecture. Architects and engineers will follow an implementation journey based on techniques and architectures that have proven to work for microservices systems. You'll build an operating model, a microservices design, an infrastructure foundation, and two working microservices, then put those pieces

together as a single implementation. For anyone tasked with building microservices or a microservices architecture, this guide is invaluable. Learn an effective and explicit end-to-end microservices system design Define teams, their responsibilities, and guidelines for working together Understand how to slice a big application into a collection of microservices Examine how to isolate and embed data into corresponding microservices Build a simple yet powerful CI/CD pipeline for infrastructure changes Write code for sample microservices Deploy a working microservices application on Amazon Web Services

Designing Microservices Platforms with NATS Packt Publishing Ltd

Understand the key challenges and solutions around building microservices in the enterprise application environment. This book provides a comprehensive understanding of microservices architectural principles and how to use microservices in real-world scenarios. Architectural challenges using microservices with service integration and API management are presented and you learn how to eliminate the use of centralized integration products such as the enterprise service bus (ESB) through the use of composite/integration microservices. Concepts in the book are supported with use cases, and emphasis is put on the reality that most of you are implementing in a “brownfield” environment in which you must implement microservices alongside legacy applications with minimal disruption to your business. Microservices for the Enterprise covers state-of-the-art techniques around microservices messaging, service development and description, service

discovery, governance, and data management technologies and guides you through the microservices design process. Also included is the importance of organizing services as core versus atomic, composite versus integration, and API versus edge, and how such organization helps to eliminate the use of a central ESB and expose services through an API gateway. What You'll Learn Design and develop microservices architectures with confidence Put into practice the most modern techniques around messaging technologies Apply the Service Mesh pattern to overcome inter-service communication challenges Apply battle-tested microservices security patterns to address real-world scenarios Handle API management, decentralized data management, and observability Who This Book Is For Developers and DevOps engineers responsible for implementing applications around a microservices architecture, and architects and analysts who are designing such systems

Reactive Systems in Java Maarten Van Steen

Learn how to build, test, secure, deploy, and efficiently consume services across distributed systems. Key Features - Explore the wealth of options provided by Spring Cloud for wiring service dependencies in microservice systems. - Create microservices utilizing Spring Cloud's Netflix OSS - Architect your cloud-native data using Spring Cloud. Book Description Developing, deploying, and operating cloud applications should be as easy as local applications. This should be the governing principle behind any cloud platform, library, or tool. Spring Cloud—an open-source library—makes it easy to develop JVM applications for the cloud. In this book, you will be introduced to Spring Cloud

and will master its features from the application developer's point of view. This book begins by introducing you to microservices for Spring and the available feature set in Spring Cloud. You will learn to configure the Spring Cloud server and run the Eureka server to enable service registration and discovery. Then you will learn about techniques related to load balancing and circuit breaking and utilize all features of the Feign client. The book now delves into advanced topics where you will learn to implement distributed tracing solutions for Spring Cloud and build message-driven microservice architectures. Before running an application on Docker containers, you will master testing and securing techniques with Spring Cloud. What you will learn - Abstract Spring Cloud's feature set - Create microservices utilizing Spring Cloud's Netflix OSS - Create synchronous API microservices based on a message-driven architecture. - Explore advanced topics such as distributed tracing, security, and contract testing. - Manage and deploy applications on the production environment

Who this book is for This book appeals to developers keen to take advantage of Spring Cloud, an open source library which helps developers quickly build distributed systems. Knowledge of Java and Spring Framework will be helpful, but no prior exposure to Spring Cloud is required.

[Microservices with Azure](#) Packt Publishing Ltd

"Domain-Driven Design" incorporates numerous examples in Java-case studies taken from actual projects that illustrate the application of domain-driven design to real-world software development.

[Microservices Patterns](#) Springer Science & Business Media

"A comprehensive overview of the challenges teams face when moving to microservices, with industry-tested solutions to these problems." - Tim Moore, Lightbend

44 reusable patterns to develop and deploy reliable production-quality microservices-based applications, with worked examples in Java

Key Features

44 design patterns for building and deploying microservices applications

Drawing on decades of unique experience from author and microservice architecture pioneer Chris Richardson

A pragmatic approach to the benefits and the drawbacks of microservices architecture

Solve service decomposition, transaction management, and inter-service communication

Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

About The Book

Microservices Patterns teaches you 44 reusable patterns to reliably develop and deploy production-quality microservices-based applications. This invaluable set of design patterns builds on decades of distributed system experience, adding new patterns for composing services into systems that scale and perform under real-world conditions. More than just a patterns catalog, this practical guide with worked examples offers industry-tested advice to help you design, implement, test, and deploy your microservices-based application. What You Will Learn

How (and why!) to use microservices architecture

Service decomposition strategies

Transaction management and querying patterns

Effective testing strategies

Deployment patterns

This Book Is Written For

Written for enterprise developers familiar with standard enterprise application architecture. Examples are in Java.

About The Author

Chris Richardson is a

Java Champion, a JavaOne rock star, author of Manning's POJOs in Action, and creator of the original CloudFoundry.com. Table of Contents Escaping monolithic hell Decomposition strategies Interprocess communication in a microservice architecture Managing transactions with sagas Designing business logic in a microservice architecture Developing business logic with event sourcing Implementing queries in a microservice architecture External API patterns Testing microservices: part 1 Testing microservices: part 2 Developing production-ready services Deploying microservices Refactoring to microservices

Building Microservices "O'Reilly Media, Inc."

One of the biggest challenges for organizations that have adopted microservice architecture is the lack of architectural, operational, and organizational standardization. After splitting a monolithic application or building a microservice ecosystem from scratch, many engineers are left wondering what's next. In this practical book, author Susan Fowler presents a set of microservice standards in depth, drawing from her experience standardizing over a thousand microservices at Uber. You'll learn how to design microservices that are stable, reliable, scalable, fault tolerant, performant, monitored, documented, and prepared for any catastrophe. Explore production-readiness standards, including: Stability and Reliability: develop, deploy, introduce, and deprecate microservices; protect against dependency failures Scalability and Performance: learn essential components for achieving greater microservice efficiency Fault Tolerance

and Catastrophe Preparedness: ensure availability by actively pushing microservices to fail in real time Monitoring: learn how to monitor, log, and display key metrics; establish alerting and on-call procedures Documentation and Understanding: mitigate tradeoffs that come with microservice adoption, including organizational sprawl and technical debt

Building Event-Driven Microservices "O'Reilly Media, Inc."

Data is at the center of many challenges in system design today. Difficult issues need to be figured out, such as scalability, consistency, reliability, efficiency, and maintainability. In addition, we have an overwhelming variety of tools, including relational databases, NoSQL datastores, stream or batch processors, and message brokers. What are the right choices for your application? How do you make sense of all these buzzwords? In this practical and comprehensive guide, author Martin Kleppmann helps you navigate this diverse landscape by examining the pros and cons of various technologies for processing and storing data. Software keeps changing, but the fundamental principles remain the same. With this book, software engineers and architects will learn how to apply those ideas in practice, and how to make full use of data in modern applications. Peer under the hood of the systems you already use, and learn how to use and operate them more effectively Make informed decisions by identifying the strengths and weaknesses of different tools Navigate the trade-offs around consistency, scalability, fault tolerance, and complexity Understand the distributed systems research upon which modern databases are built Peek behind the scenes of major online services, and

learn from their architectures

Related with Building Distributed Systems And Microservices In Go With:

- Force And Acceleration Worksheet Answer Key : [click here](#)