

---

# Read Free Building Microservices Designing Fine Grained Systems

---

When somebody should go to the book stores, search establishment by shop, shelf by shelf, it is in reality problematic. This is why we allow the books compilations in this website. It will categorically ease you to look guide **Building Microservices Designing Fine Grained Systems** as you such as.

By searching the title, publisher, or authors of guide you in point of fact want, you can discover them rapidly. In the house, workplace, or perhaps in your method can be every best area within net connections. If you wish to download and install the Building Microservices Designing Fine Grained Systems, it is definitely easy then, since currently we extend the belong to to buy and make bargains to download and install Building Microservices Designing Fine Grained Systems therefore simple!

---

**KEY=MICROSERVICES - RILEY BENTON**

---

**BUILDING MICROSERVICES**

---

*Oreilly & Associates Incorporated* **Over the past 10 years, distributed systems have become more fine-grained. From the large multi-million line long monolithic applications, we are now seeing the benefits of smaller self-contained services. Rather than heavy-weight, hard to change Service Oriented Architectures, we are now seeing systems consisting of collaborating microservices. Easier to change, deploy, and if required retire, organizations which are in the right position to take advantage of them are yielding significant benefits. This book takes an holistic view of the things you need to be cognizant of in order to pull this off. It covers just enough understanding of technology, architecture, operations and organization to show you how to move towards finer-grained systems.**

---

**BUILDING MICROSERVICES**

---

---

## DESIGNING FINE-GRAINED SYSTEMS

---

*"O'Reilly Media, Inc."* **Annotation** Over the past 10 years, distributed systems have become more fine-grained. From the large multi-million line long monolithic applications, we are now seeing the benefits of smaller self-contained services. Rather than heavy-weight, hard to change Service Oriented Architectures, we are now seeing systems consisting of collaborating microservices. Easier to change, deploy, and if required retire, organizations which are in the right position to take advantage of them are yielding significant benefits. This book takes an holistic view of the things you need to be cognizant of in order to pull this off. It covers just enough understanding of technology, architecture, operations and organization to show you how to move towards finer-grained systems.

---

---

## MONOLITH TO MICROSERVICES

---

---

### EVOLUTIONARY PATTERNS TO TRANSFORM YOUR MONOLITH

---

*O'Reilly Media* **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**

---

---

## PRODUCTION-READY MICROSERVICES

---

---

## **BUILDING STANDARDIZED SYSTEMS ACROSS AN ENGINEERING ORGANIZATION**

---

*"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* **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**

---

## MICROSERVICE ARCHITECTURE

---

---

### ALIGNING PRINCIPLES, PRACTICES, AND CULTURE

---

*"O'Reilly Media, Inc."* **Microservices can have a positive impact on your enterprise—just ask Amazon and Netflix—but you can fall into many traps if you don't approach them in the right way. This practical guide covers the entire microservices landscape, including the principles, technologies, and methodologies of this unique, modular style of system building. You'll learn about the experiences of organizations around the globe that have successfully adopted microservices. In three parts, this book explains how these services work and what it means to build an application the Microservices Way. You'll explore a design-based approach to microservice architecture with guidance for implementing various elements. And you'll get a set of recipes and practices for meeting practical, organizational, and cultural challenges to microservice adoption. Learn how microservices can help you drive business objectives Examine the principles, practices, and culture that define microservice architectures Explore a model for creating complex systems and a design process for building a microservice architecture Learn the fundamental design concepts for individual microservices Delve into the operational elements of a microservices architecture, including containers and service discovery Discover how to handle the challenges of introducing microservice architecture in your organization**

---

### MICROSERVICES: UP AND RUNNING

---

*O'Reilly Media* **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 DISTRIBUTED SYSTEMS

---

---

### PATTERNS AND PARADIGMS FOR SCALABLE, RELIABLE SERVICES

---

*"O'Reilly Media, Inc."* In the race to compete in today's fast-moving markets, large enterprises are busy adopting new technologies for creating new products, processes, and business models. But one obstacle on the road to digital transformation is placing too much emphasis on technology, and not enough on the types of processes technology enables. What if different lines of business could build their own services and applications—and decision-making was distributed rather than centralized? This report explores the concept of a digital business platform as a way of empowering individual business sectors to act on data in real time. Much innovation in a digital enterprise will increasingly happen at the edge, whether it involves business users (from marketers to data scientists) or IoT devices. To facilitate the process, your core IT team can provide these sectors with the digital tools they need to innovate quickly. This report explores: Key cultural and organizational changes for developing business capabilities through cross-functional product teams A platform for integrating applications, data sources, business partners, clients, mobile apps, social networks, and IoT devices Creating internal API programs for building innovative edge services in low-code or no-code environments Tools including Integration Platform as a Service, Application Platform as a Service, and Integration Software as a Service The challenge of integrating microservices and serverless architectures Event-driven architectures for processing and reacting to events in real time You'll also learn about a complete pervasive integration solution as a core component of a digital business platform to serve every audience in your organization.

---

### BUILDING SECURE AND RELIABLE SYSTEMS

---

---

### BEST PRACTICES FOR DESIGNING, IMPLEMENTING, AND MAINTAINING SYSTEMS

---

*O'Reilly Media* Can a system be considered truly reliable if it isn't fundamentally secure? Or can it be considered secure if it's unreliable? Security is crucial to the design and operation of scalable systems in production, as it plays an important part in product quality, performance, and availability. In this book, experts from Google share best practices to help your organization design scalable and reliable systems that are fundamentally secure. Two previous O'Reilly

books from Google—Site Reliability Engineering and The Site Reliability Workbook—demonstrated how and why a commitment to the entire service lifecycle enables organizations to successfully build, deploy, monitor, and maintain software systems. In this latest guide, the authors offer insights into system design, implementation, and maintenance from practitioners who specialize in security and reliability. They also discuss how building and adopting their recommended best practices requires a culture that's supportive of such change. You'll learn about secure and reliable systems through: Design strategies Recommendations for coding, testing, and debugging practices Strategies to prepare for, respond to, and recover from incidents Cultural best practices that help teams across your organization collaborate effectively

---

## **SOFTWARE ARCHITECTURE: THE HARD PARTS**

---

*"O'Reilly Media, Inc."* There are no easy decisions in software architecture. Instead, there are many hard parts--difficult problems or issues with no best practices--that force you to choose among various compromises. With this book, you'll learn how to think critically about the trade-offs involved with distributed architectures. Architecture veterans and practicing consultants Neal Ford, Mark Richards, Pramod Sadalage, and Zhamak Dehghani discuss strategies for choosing an appropriate architecture. By interweaving a story about a fictional group of technology professionals--the Sysops Squad--they examine everything from how to determine service granularity, manage workflows and orchestration, manage and decouple contracts, and manage distributed transactions to how to optimize operational characteristics, such as scalability, elasticity, and performance. By focusing on commonly asked questions, this book provides techniques to help you discover and weigh the trade-offs as you confront the issues you face as an architect. Analyze trade-offs and effectively document your decisions Make better decisions regarding service granularity Understand the complexities of breaking apart monolithic applications Manage and decouple contracts between services Handle data in a highly distributed architecture Learn patterns to manage workflow and transactions when breaking apart applications

---

## **MICROSERVICES FOR THE ENTERPRISE**

---



---

### **DESIGNING, DEVELOPING, AND DEPLOYING**

---

*Apress* 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

---

## **SECURITY AND MICROSERVICE ARCHITECTURE ON AWS**

---

*"O'Reilly Media, Inc."* Security is usually an afterthought when organizations design microservices for cloud systems. Most companies today are exposed to potential security threats, but their response is more reactive than proactive. That leads to unnecessarily complicated architecture that's harder to implement and even harder to manage and scale. Author Gaurav Raje shows you how to build highly secure systems on AWS without increasing overhead. Ideal for cloud solution architects and software developers with AWS experience, this practical book starts with a high-level architecture and design discussion, then explains how to implement your solution in the cloud in a secure but frictionless manner. By leveraging the AWS Shared Responsibility Model, you'll be able to: Achieve complete mediation in microservices at the infrastructure level Implement a secure and reliable audit trail of all events within the system Develop architecture that aims to simplify compliance with various regulations in finance, medicine, and legal services Put systems in place that detect anomalous behavior and alert the proper administrators in case of a breach Scale security mechanisms on individual microservices independent of each other.

---

## ARCHITECTING FOR SCALE

---

---

### HOW TO MAINTAIN HIGH AVAILABILITY AND MANAGE RISK IN THE CLOUD

---

*O'Reilly Media* Every day, companies struggle to scale critical applications. As traffic volume and data demands increase, these applications become more complicated and brittle, exposing risks and compromising availability. With the popularity of software as a service, scaling has never been more important. Updated with an expanded focus on modern architecture paradigms such as microservices and cloud computing, this practical guide provides techniques for building systems that can handle huge quantities of traffic, data, and demand—without affecting the quality your customers expect. Architects, managers, and directors in engineering and operations organizations will learn how to build applications at scale that run more smoothly and reliably to meet the needs of customers. Learn how scaling affects the availability of your services, why that matters, and how to improve it Dive into a modern service-based application architecture that ensures high availability and reduces the effects of service failures Explore the Single Team Owned Service Architecture paradigm (STOSA)—a model for scaling your development organization in tandem with your application Understand, measure, and mitigate risk in your systems Use the cloud to build highly scalable applications

---

## FUNDAMENTALS OF SOFTWARE ARCHITECTURE

---

---

### AN ENGINEERING APPROACH

---

*O'Reilly Media* Salary surveys worldwide regularly place software architect in the top 10 best jobs, yet no real guide exists to help developers become architects. Until now. This book provides the first comprehensive overview of software architecture's many aspects. Aspiring and existing architects alike will examine architectural characteristics, architectural patterns, component determination, diagramming and presenting architecture, evolutionary architecture, and many other topics. Mark Richards and Neal Ford—hands-on practitioners who have taught software architecture classes professionally for years—focus on architecture principles that apply across all technology stacks. You'll explore software architecture in a modern light, taking into account all the innovations of the past decade. This book examines: Architecture patterns: The technical basis for many architectural decisions Components: Identification, coupling, cohesion, partitioning, and granularity Soft skills: Effective team management, meetings, negotiation,

presentations, and more Modernity: Engineering practices and operational approaches that have changed radically in the past few years Architecture as an engineering discipline: Repeatable results, metrics, and concrete valuations that add rigor to software architecture

---

## **BUILDING MICROSERVICES WITH ASP.NET CORE**

---

---

### **DEVELOP, TEST, AND DEPLOY CROSS-PLATFORM SERVICES IN THE CLOUD**

---

*"O'Reilly Media, Inc."* At a time when nearly every vertical, regardless of domain, seems to need software running in the cloud to make money, microservices provide the agility and drastically reduced time to market you require. This hands-on guide shows you how to create, test, compile, and deploy microservices, using the ASP.NET Core free and open-source framework. Along the way, you'll pick up good, practical habits for building powerful and robust services. Building microservices isn't about learning a specific framework or programming language; it's about building applications that thrive in elastically scaling environments that don't have host affinity, and that can start and stop at a moment's notice. This practical book guides you through the process. Learn test-driven and API-first development concepts Communicate with other services by creating and consuming backing services such as databases and queues Build a microservice that depends on an external data source Learn about event sourcing, the event-centric approach to persistence Use ASP.NET Core to build web applications designed to thrive in the cloud Build a service that consumes, or is consumed by, other services Create services and applications that accept external configuration Explore ways to secure ASP.NET Core microservices and applications

---

## **MICROSERVICES: PATTERNS AND APPLICATIONS**

---

---

### **DESIGNING FINE-GRAINED SERVICES BY APPLYING PATTERNS**

---

Microservices: Patterns and Applications Microservices are the next big thing in designing scalable, easy to maintain applications. This book will explain everything you need to know about Microservices to make your next project successful. You will learn: Microservice Patterns This book goes into great detail on all of the Microservice Architecture patterns including \* Monolithic Architecture\* Microservice Architecture\* Service Discovery\* Gateway / Proxy API\* Orchestrated API\* Service Registration\* CQRS and Event Sourcing\* Bulk Heads\* Circuit Breaker\* Message Broker The most important thing about Microservices is when and how to apply a pattern, along with explaining what choices you

must make and why. Every system is different so it is vital to understand a lot of basics before designing and developing your own Microservices. From Monolithic to Microservice The basics here are how to decompose a Monolithic system into a Microservice and this book shows exactly how this process is completed. Service Oriented Architecture to Microservice A more common need is to migrate your system from a SOA based architecture to Microservices, there are many advantages and the process is not as straightforward as you would expect. New Microservices If you want to build a brand-new system and leverage the power of Microservices this book outlines the pitfalls, strategies and tactics needs to make this work for you. It is not as easy as it would seem and you will understand why after reading this book. Microservice Technologies You'll learn about what technologies you need to use and understand for successful Microservices. \*Virtualization\*Containers (Docker and Rocket)\*Databases\*Security (JSON Web Tokens)\*Logging\*Exceptions\*Caching\*Timeouts\*Scalability (CAP, Cube)\*Platform as a Service (PaaS)\*Cloud architecture\*Technology agnostic Why Microservices? Isn't this just the latest buzz word? While Microservices may be a recent trend and is gaining traction across the industry as a silver-bullet. It is not a silver-bullet. In this book you will learn important reasons why you cannot treat Microservices or any technology or technique as a silver-bullet. There are tradeoffs and advantages to every architectural decision, you will understand the details by reading this book. Most importantly you will understand how Microservices is what SOA had promised and never delivered. Author: Lucas Krause Lucas has been in the technology industry as a consultant, contractor, architect, engineer, and manager and understands and has used Microservices successfully to solve his client problems. Philosophy of Microservices You'll learn about what the philosophy of Microservices is and why this is important. It is critical to understand the philosophy as that is what makes Microservices work at so many other companies and solutions. If you are looking to gain an understanding of Microservices along with the patterns and application around the process to implementing them than, this is the book for you! Ready to learn about Microservices? Let's go! Want To Be brought up to speed on the latest innovations and techniques with Microservices? Want to Understand Why Microservices? What Makes Microservices so Special? What are the potential pitfalls? Why Are Microservices so popular? How do I make my projects successful?

---

## **BUILDING MICRO-FRONTENDS**

---

"O'Reilly Media, Inc." What's the answer to today's increasingly complex web applications? Micro-frontends. Inspired by the microservices model, this approach lets you break interfaces into separate features managed by different teams of developers. With this practical guide, Luca Mezzalana shows software architects, tech leads, and software developers

how to build and deliver artifacts atomically rather than use a big bang deployment. You'll learn how micro-frontends enable your team to choose any library or framework. This gives your organization technical flexibility and allows you to hire and retain a broad spectrum of talent. Micro-frontends also support distributed or colocated teams more efficiently. Pick up this book and learn how to get started with this technological breakthrough right away. Explore available frontend development architectures Learn how microservice principles apply to frontend development Understand the four pillars for creating a successful micro-frontend architecture Examine the benefits and pitfalls of existing micro-frontend architectures Learn principles and best practices for creating successful automation strategies Discover patterns for integrating micro-frontend architectures using microservices or a monolith API layer

---

## **MICROSERVICES IN .NET CORE**

---

### **WITH C#, THE NANCY FRAMEWORK, AND OWIN MIDDLEWARE**

---

*Manning Publications* **Microservices are responsible for very tightly focused capabilities that are part of a more complex server-side software system. Microservices, when done well, are malleable, scalable, resilient, and allow a short lead time from start of implementation to deployment to production. When using microservices, the need for the technology to be lightweight and low ceremony grows, because creating new microservices needs to be quick and easy. OWIN is great for reuse of plumbing code and a lightweight web framework, like Nancy, is ideal. Microservices in .NET Core teaches readers how to build and deploy secure and operations-friendly microservices using Nancy. The book starts with an introduction to the microservices architectural style. Next, readers learn important practical aspects of developing microservices from simple core concepts to more sophisticated. Throughout the book, readers will see many code examples implementing it with lightweight .NET technologies' most prominently Nancy. By the end, they'll be able to quickly and easily build reliable and operations-friendly microservices using Nancy, OWIN and other open technologies. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.**

---

## **MICROSERVICES IN ACTION**

---

*Simon and Schuster* **Summary Microservices in Action is a practical book about building and deploying microservice-based applications. Written for developers and architects with a solid grasp of service-oriented development, it tackles the challenge of putting microservices into production. Purchase of the print book includes a free eBook in PDF, Kindle,**

and ePub formats from Manning Publications. About the Technology Invest your time in designing great applications, improving infrastructure, and making the most out of your dev teams. Microservices are easier to write, scale, and maintain than traditional enterprise applications because they're built as a system of independent components. Master a few important new patterns and processes, and you'll be ready to develop, deploy, and run production-quality microservices. About the Book Microservices in Action teaches you how to write and maintain microservice-based applications. Created with day-to-day development in mind, this informative guide immerses you in real-world use cases from design to deployment. You'll discover how microservices enable an efficient continuous delivery pipeline, and explore examples using Kubernetes, Docker, and Google Container Engine. What's inside An overview of microservice architecture Building a delivery pipeline Best practices for designing multi-service transactions and queries Deploying with containers Monitoring your microservices About the Reader Written for intermediate developers familiar with enterprise architecture and cloud platforms like AWS and GCP. About the Author Morgan Bruce and Paulo A. Pereira are experienced engineering leaders. They work daily with microservices in a production environment, using the techniques detailed in this book. Table of Contents PART 1 - The lay of the land Designing and running microservices Microservices at SimpleBank PART 2 - Design Architecture of a microservice application Designing new features Transactions and queries in microservices Designing reliable services Building a reusable microservice framework PART 3 - Deployment Deploying microservices Deployment with containers and schedulers Building a delivery pipeline for microservices PART 4 - Observability and ownership Building a monitoring system Using logs and traces to understand behavior Building microservice teams

---

## **DESIGNING MICROSERVICES PLATFORMS WITH NATS**

---

### **A MODERN APPROACH TO DESIGNING AND IMPLEMENTING SCALABLE MICROSERVICES PLATFORMS WITH NATS MESSAGING**

---

*Packt Publishing Ltd* A complete reference for designing and building scalable microservices platforms with NATS messaging technology for inter-service communication with security and observability Key Features Understand the use of a messaging backbone for inter-service communication in microservices architecture Design and build a real-world microservices platform with NATS as the messaging backbone using the Go programming language Explore security, observability, and best practices for building a microservices platform with NATS Book Description Building a scalable microservices platform that caters to business demands is critical to the success of that platform. In a

microservices architecture, inter-service communication becomes a bottleneck when the platform scales. This book provides a reference architecture along with a practical example of how to implement it for building microservices-based platforms with NATS as the messaging backbone for inter-service communication. In *Designing Microservices Platforms with NATS*, you'll learn how to build a scalable and manageable microservices platform with NATS. The book starts by introducing concepts relating to microservices architecture, inter-service communication, messaging backbones, and the basics of NATS messaging. You'll be introduced to a reference architecture that uses these concepts to build a scalable microservices platform and guided through its implementation. Later, the book touches on important aspects of platform securing and monitoring with the help of the reference implementation. Finally, the book concludes with a chapter on best practices to follow when integrating with existing platforms and the future direction of microservices architecture and NATS messaging as a whole. By the end of this microservices book, you'll have developed the skills to design and implement microservices platforms with NATS. What you will learn

Understand the concepts of microservices architecture  
Get to grips with NATS messaging technology  
Handle transactions and message delivery guarantees with microservices  
Implement a reference architecture for microservices using NATS  
Discover how to improve the platform's security and observability  
Explore how a NATS microservices platform integrates with an enterprise ecosystem

Who this book is for  
This book is for enterprise software architects and developers who want to gain hands-on microservices experience for designing, implementing, and managing complex distributed systems with microservices architecture concepts. Intermediate-level experience in any programming language and software architecture is required to make the most of this book.

---

## **BUILDING EVOLUTIONARY ARCHITECTURES**

---

### **SUPPORT CONSTANT CHANGE**

---

*"O'Reilly Media, Inc."* The software development ecosystem is constantly changing, providing a constant stream of new tools, frameworks, techniques, and paradigms. Over the past few years, incremental developments in core engineering practices for software development have created the foundations for rethinking how architecture changes over time, along with ways to protect important architectural characteristics as it evolves. This practical guide ties those parts together with a new way to think about architecture and time.

---

## EMBRACING MICROSERVICES DESIGN

---

---

### A PRACTICAL GUIDE TO REVEALING ANTI-PATTERNS AND ARCHITECTURAL PITFALLS TO AVOID MICROSERVICES FALLACIES

---

*Packt Publishing Ltd* **Develop microservice-based enterprise applications with expert guidance to avoid failures and technological debt with the help of real-world examples** **Key Features** Implement the right microservices adoption strategy to transition from monoliths to microservices Explore real-world use cases that explain anti-patterns and alternative practices in microservices development Discover proven recommendations for avoiding architectural mistakes when designing microservices **Book Description** Microservices have been widely adopted for designing distributed enterprise apps that are flexible, robust, and fine-grained into services that are independent of each other. There has been a paradigm shift where organizations are now either building new apps on microservices or transforming existing monolithic apps into microservices-based architecture. This book explores the importance of anti-patterns and the need to address flaws in them with alternative practices and patterns. You'll identify common mistakes caused by a lack of understanding when implementing microservices and cover topics such as organizational readiness to adopt microservices, domain-driven design, and resiliency and scalability of microservices. The book further demonstrates the anti-patterns involved in re-platforming brownfield apps and designing distributed data architecture. You'll also focus on how to avoid communication and deployment pitfalls and understand cross-cutting concerns such as logging, monitoring, and security. Finally, you'll explore testing pitfalls and establish a framework to address isolation, autonomy, and standardization. By the end of this book, you'll have understood critical mistakes to avoid while building microservices and the right practices to adopt early in the product life cycle to ensure the success of a microservices initiative. **What you will learn** Discover the responsibilities of different individuals involved in a microservices initiative Avoid the common mistakes in architecting microservices for scalability and resiliency Understand the importance of domain-driven design when developing microservices Identify the common pitfalls involved in migrating monolithic applications to microservices Explore communication strategies, along with their potential drawbacks and alternatives Discover the importance of adopting governance, security, and monitoring Understand the role of CI/CD and testing **Who this book is for** This practical microservices book is for software architects, solution architects, and developers involved in designing microservices architecture and its development, who want to gain insights into avoiding pitfalls and drawbacks in distributed applications, and save time

and money that might otherwise get wasted if microservices designs fail. Working knowledge of microservices is assumed to get the most out of this book.

---

## REST IN PRACTICE

---

---

## HYPERMEDIA AND SYSTEMS ARCHITECTURE

---

*"O'Reilly Media, Inc."* **REST continues to gain momentum as the best method for building Web services, and this down-to-earth book delivers techniques and examples that show how to design and implement integration solutions using the REST architectural style.**

---

## LEARNING DOMAIN-DRIVEN DESIGN

---

*"O'Reilly Media, Inc."* **Building software is harder than ever. As a developer, you not only have to chase ever-changing technological trends but also need to understand the business domains behind the software. This practical book provides you with a set of core patterns, principles, and practices for analyzing business domains, understanding business strategy, and, most importantly, aligning software design with its business needs. Author Vlad Khononov shows you how these practices lead to robust implementation of business logic and help to future-proof software design and architecture. You'll examine the relationship between domain-driven design (DDD) and other methodologies to ensure you make architectural decisions that meet business requirements. You'll also explore the real-life story of implementing DDD in a startup company. With this book, you'll learn how to: Analyze a company's business domain to learn how the system you're building fits its competitive strategy Use DDD's strategic and tactical tools to architect effective software solutions that address business needs Build a shared understanding of the business domains you encounter Decompose a system into bounded contexts Coordinate the work of multiple teams Gradually introduce DDD to brownfield projects**

---

## REST API DESIGN RULEBOOK

---

*"O'Reilly Media, Inc."* **The basic rules of REST APIs - "many nouns, few verbs, stick with HTTP" - seem easy, but that simplicity and power require discipline to work smoothly. This brief guide provides next steps for implementing complex projects on simple and extensible foundations.**

---



---

## MICROSERVICES PATTERNS

---

### WITH EXAMPLES IN JAVA

---

*Simon and Schuster* "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](https://www.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

---



---

## MICROSERVICES

---

---

## A PRACTICAL GUIDE

---

*Createspace Independent Publishing Platform* **Microservices have many advantages: Efficiently implementing more features, bringing software into production faster, robustness and easy scalability are among them. But implementing a microservices architecture and selecting the necessary technologies are difficult challenges. This book shows microservices recipes that architects can customize and combine into a microservices menu. In this way, the implementation of microservices can be individually adapted to the requirements of the project. Eberhard Wolff introduces microservices, self-contained systems, micro- and macro-architecture and the migration to microservices. The second part shows the microservices recipes: Basic technologies such as Docker or PaaS, frontend integration with links, JavaScript or ESI (Edge Side Includes). This is followed by asynchronous microservices with Apache Kafka or REST / Atom. In the synchronous approaches, the book discusses REST with the Netflix stack, Consul, PaaS with Cloud Foundry, and Kubernetes. Finally, operations is discussed: Log Analysis with Elasticsearch and Kibana, Monitoring with Prometheus, and tracing with Zipkin. For each recipe there are suggestions for variations and combinations. Readers can experience all technologies hands-on with a demo project on GitHub. The outlook picks up on the operation of microservices and also shows how the reader can start with microservices in concrete terms. The book provides the technical tools to implement a microservices architecture. Demo projects and suggestions for self-study will complete the book.**

---

## BOOTSTRAPPING MICROSERVICES WITH DOCKER, KUBERNETES, AND TERRAFORM

---

### A PROJECT-BASED GUIDE

---

*Manning Publications* **Summary The best way to learn microservices development is to build something! Bootstrapping Microservices with Docker, Kubernetes, and Terraform guides you from zero through to a complete microservices project, including fast prototyping, development, and deployment. You'll get your feet wet using industry-standard tools as you learn and practice the practical skills you'll use for every microservices application. Following a true bootstrapping approach, you'll begin with a simple, familiar application and build up your knowledge and skills as you create and deploy a real microservices project. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the technology Taking microservices from proof of concept to production is a complex, multi-step operation relying on tools like Docker, Terraform, and Kubernetes for packaging**

and deployment. The best way to learn the process is to build a project from the ground up, and that's exactly what you'll do with this book! About the book *In Bootstrapping Microservices with Docker, Kubernetes, and Terraform*, author Ashley Davis lays out a comprehensive approach to building microservices. You'll start with a simple design and work layer-by-layer until you've created your own video streaming application. As you go, you'll learn to configure cloud infrastructure with Terraform, package microservices using Docker, and deploy your finished project to a Kubernetes cluster. What's inside *Developing and testing microservices applications Working with cloud providers Applying automated testing Implementing infrastructure as code and setting up a continuous delivery pipeline Monitoring, managing, and troubleshooting About the reader Examples are in JavaScript. No experience with microservices, Kubernetes, Terraform, or Docker required. About the author Ashley Davis is a software developer, entrepreneur, stock trader, and the author of Manning's Data Wrangling with JavaScript. Table of Contents 1 Why microservices? 2 Creating your first microservice 3 Publishing your first microservice 4 Data management for microservices 5 Communication between microservices 6 Creating your production environment 7 Getting to continuous delivery 8 Automated testing for microservices 9 Exploring FlixTube 10 Healthy microservices 11 Pathways to scalability*

---

**RELEASE IT!**

---

## **DESIGN AND DEPLOY PRODUCTION-READY SOFTWARE**

---

*Pragmatic Bookshelf* A single dramatic software failure can cost a company millions of dollars - but can be avoided with simple changes to design and architecture. This new edition of the best-selling industry standard shows you how to create systems that run longer, with fewer failures, and recover better when bad things happen. New coverage includes DevOps, microservices, and cloud-native architecture. Stability antipatterns have grown to include systemic problems in large-scale systems. This is a must-have pragmatic guide to engineering for production systems. If you're a software developer, and you don't want to get alerts every night for the rest of your life, help is here. With a combination of case studies about huge losses - lost revenue, lost reputation, lost time, lost opportunity - and practical, down-to-earth advice that was all gained through painful experience, this book helps you avoid the pitfalls that cost companies millions of dollars in downtime and reputation. Eighty percent of project life-cycle cost is in production, yet few books address this topic. This updated edition deals with the production of today's systems - larger, more complex, and heavily virtualized - and includes information on chaos engineering, the discipline of

applying randomness and deliberate stress to reveal systematic problems. Build systems that survive the real world, avoid downtime, implement zero-downtime upgrades and continuous delivery, and make cloud-native applications resilient. Examine ways to architect, design, and build software - particularly distributed systems - that stands up to the typhoon winds of a flash mob, a Slashdotting, or a link on Reddit. Take a hard look at software that failed the test and find ways to make sure your software survives. To skip the pain and get the experience...get this book.

---

## **INFRASTRUCTURE AS CODE**

---

*O'Reilly Media* Six years ago, Infrastructure as Code was a new concept. Today, as even banks and other conservative organizations plan moves to the cloud, development teams for companies worldwide are attempting to build large infrastructure codebases. With this practical book, Kief Morris of ThoughtWorks shows you how to effectively use principles, practices, and patterns pioneered by DevOps teams to manage cloud-age infrastructure. Ideal for system administrators, infrastructure engineers, software developers, team leads, and architects, this updated edition demonstrates how you can exploit cloud and automation technology to make changes easily, safely, quickly, and responsibly. You'll learn how to define everything as code and apply software design and engineering practices to build your system from small, loosely coupled pieces. This book covers: Foundations: Use Infrastructure as Code to drive continuous change and raise the bar of operational quality, using tools and technologies to build cloud-based platforms Working with infrastructure stacks: Learn how to define, provision, test, and continuously deliver changes to infrastructure resources Working with servers and other platforms: Use patterns to design provisioning and configuration of servers and clusters Working with large systems and teams: Learn workflows, governance, and architectural patterns to create and manage infrastructure elements

---

## **THE ART OF SCALABILITY**

---

---

### **SCALABLE WEB ARCHITECTURE, PROCESSES, AND ORGANIZATIONS FOR THE MODERN ENTERPRISE**

---

*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

---

## RESTFUL WEB APIS

---

## SERVICES FOR A CHANGING WORLD

---

*"O'Reilly Media, Inc."* The popularity of REST in recent years has led to tremendous growth in almost-RESTful APIs that don't include many of the architecture's benefits. With this practical guide, you'll learn what it takes to design usable REST APIs that evolve over time. By focusing on solutions that cross a variety of domains, this book shows you how to create powerful and secure applications, using the tools designed for the world's most successful distributed computing system: the World Wide Web. You'll explore the concepts behind REST, learn different strategies for creating hypermedia-based APIs, and then put everything together with a step-by-step guide to designing a RESTful Web API. Examine API design strategies, including the collection pattern and pure hypermedia Understand how hypermedia ties representations together into a coherent API Discover how XMDP and ALPS profile formats can help you meet the Web API "semantic challenge" Learn close to two-dozen standardized hypermedia data formats Apply best practices for using HTTP in API implementations Create Web APIs with the JSON-LD standard and other the Linked Data approaches Understand the CoAP protocol for using REST in embedded systems

---

## FLOW ARCHITECTURES

---

*O'Reilly Media* Software development today is embracing events and streaming data, which optimizes not only how

technology interacts but also how businesses integrate with one another to meet customer needs. This phenomenon, called flow, consists of patterns and standards that determine which activity and related data is communicated between parties over the internet. This book explores critical implications of that evolution: What happens when events and data streams help you discover new activity sources to enhance existing businesses or drive new markets? What technologies and architectural patterns can position your company for opportunities enabled by flow? James Urquhart, global field CTO at VMware, guides enterprise architects, software developers, and product managers through the process. Learn the benefits of flow dynamics when businesses, governments, and other institutions integrate via events and data streams Understand the value chain for flow integration through Wardley mapping visualization and promise theory modeling Walk through basic concepts behind today's event-driven systems marketplace Learn how today's integration patterns will influence the real-time events flow in the future Explore why companies should architect and build software today to take advantage of flow in coming years

---

## **MICROSERVICES IN .NET, SECOND EDITION**

---

*Simon and Schuster* In *Microservices in .NET, Second Edition*, you'll start by getting to grips with the unique architectural style of microservices, then move on quickly to practical development skills for building your own microservices. *Microservices in .NET, Second Edition* is a comprehensive guide to building microservice applications using the .NET stack. After a crystal-clear introduction to the microservices architectural style, it teaches you practical microservices development skills. In *Microservices in .NET, Second Edition*, you'll start by getting to grips with the unique architectural style of microservices, then move on quickly to practical development skills for building your own microservices. You'll design and build individual services in C# and learn how to compose them into a simple but functional application back end. In brand-new coverage in this second edition of the bestselling original, you'll also learn about scoping microservices and how to handle the complexities of deploying to Kubernetes. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

---

## **PRO SPRING BOOT 2**

---

---

## **AN AUTHORITATIVE GUIDE TO BUILDING MICROSERVICES, WEB AND ENTERPRISE APPLICATIONS, AND BEST PRACTICES**

---

*Apress* Quickly and productively develop complex Spring applications and microservices out of the box, with minimal

concern over things like configurations. This revised book will show you how to fully leverage the Spring Boot 2 technology and how to apply it to create enterprise ready applications that just work. It will also cover what's been added to the new Spring Boot 2 release, including Spring Framework 5 features like WebFlux, Security, Actuator and the new way to expose Metrics through Micrometer framework, and more. This book is your authoritative hands-on practical guide for increasing your enterprise Java and cloud application productivity while decreasing development time. It's a no nonsense guide with case studies of increasing complexity throughout the book. The author, a senior solutions architect and Principal Technical instructor with Pivotal, the company behind the Spring Framework, shares his experience, insights and first-hand knowledge about how Spring Boot technology works and best practices. **Pro Spring Boot 2 is an essential book for your Spring learning and reference library. What You Will Learn** Configure and use Spring Boot Use non-functional requirements with Spring Boot Actuator Carry out web development with Spring Boot Persistence with JDBC, JPA and NoSQL Databases Messaging with JMS, RabbitMQ and WebSockets Test and deploy with Spring Boot A quick look at the Spring Cloud projects Microservices and deployment to the Cloud Extend Spring Boot by creating your own Spring Boot Starter and @Enable feature Who This Book Is For Experienced Spring and Java developers seeking increased productivity gains and decreased complexity and development time in their applications and software services.

---

## **JUST ENOUGH SOFTWARE ARCHITECTURE**

---

### **A RISK-DRIVEN APPROACH**

---

*Marshall & Brainerd* This is a practical guide for software developers, and different than other software architecture books. Here's why: It teaches risk-driven architecting. There is no need for meticulous designs when risks are small, nor any excuse for sloppy designs when risks threaten your success. This book describes a way to do just enough architecture. It avoids the one-size-fits-all process tar pit with advice on how to tune your design effort based on the risks you face. It democratizes architecture. This book seeks to make architecture relevant to all software developers. Developers need to understand how to use constraints as guiderails that ensure desired outcomes, and how seemingly small changes can affect a system's properties. It cultivates declarative knowledge. There is a difference between being able to hit a ball and knowing why you are able to hit it, what psychologists refer to as procedural knowledge versus declarative knowledge. This book will make you more aware of what you have been doing and provide names for the concepts. It emphasizes the engineering. This book focuses on the technical parts of software development and

what developers do to ensure the system works not job titles or processes. It shows you how to build models and analyze architectures so that you can make principled design tradeoffs. It describes the techniques software designers use to reason about medium to large sized problems and points out where you can learn specialized techniques in more detail. It provides practical advice. Software design decisions influence the architecture and vice versa. The approach in this book embraces drill-down/pop-up behavior by describing models that have various levels of abstraction, from architecture to data structure design.

---

## **FOWLER**

---

### **PATTERN ENTERPR APPLICA ARCH**

---

*Addison-Wesley* The practice of enterprise application development has benefited from the emergence of many new enabling technologies. Multi-tiered object-oriented platforms, such as Java and .NET, have become commonplace. These new tools and technologies are capable of building powerful applications, but they are not easily implemented. Common failures in enterprise applications often occur because their developers do not understand the architectural lessons that experienced object developers have learned. Patterns of Enterprise Application Architecture is written in direct response to the stiff challenges that face enterprise application developers. The author, noted object-oriented designer Martin Fowler, noticed that despite changes in technology--from Smalltalk to CORBA to Java to .NET--the same basic design ideas can be adapted and applied to solve common problems. With the help of an expert group of contributors, Martin distills over forty recurring solutions into patterns. The result is an indispensable handbook of solutions that are applicable to any enterprise application platform. This book is actually two books in one. The first section is a short tutorial on developing enterprise applications, which you can read from start to finish to understand the scope of the book's lessons. The next section, the bulk of the book, is a detailed reference to the patterns themselves. Each pattern provides usage and implementation information, as well as detailed code examples in Java or C#. The entire book is also richly illustrated with UML diagrams to further explain the concepts. Armed with this book, you will have the knowledge necessary to make important architectural decisions about building an enterprise application and the proven patterns for use when building them. The topics covered include · Dividing an enterprise application into layers · The major approaches to organizing business logic · An in-depth treatment of mapping between objects and relational databases · Using Model-View-Controller to organize a Web presentation · Handling concurrency for data that spans multiple transactions · Designing distributed object interfaces

---

## PEOPLEWARE

---

---

### PRODUCTIVE PROJECTS AND TEAMS

---

*Addison-Wesley* Few books in computing have had as profound an influence on software management as **Peopleware**. The unique insight of this longtime best seller is that the major issues of software development are human, not technical. They're not easy issues; but solve them, and you'll maximize your chances of success. "Peopleware has long been one of my two favorite books on software engineering. Its underlying strength is its base of immense real experience, much of it quantified. Many, many varied projects have been reflected on and distilled; but what we are given is not just lifeless distillate, but vivid examples from which we share the authors' inductions. Their premise is right: most software project problems are sociological, not technological. The insights on team jelling and work environment have changed my thinking and teaching. The third edition adds strength to strength." — Frederick P. Brooks, Jr., Kenan Professor of Computer Science, University of North Carolina at Chapel Hill, Author of *The Mythical Man-Month* and *The Design of Design* "Peopleware is the one book that everyone who runs a software team needs to read and reread once a year. In the quarter century since the first edition appeared, it has become more important, not less, to think about the social and human issues in software development. This is the only way we're going to make more humane, productive workplaces. Buy it, read it, and keep a stock on hand in the office supply closet." —Joel Spolsky, Co-founder, Stack Overflow "When a book about a field as volatile as software design and use extends to a third edition, you can be sure that the authors write of deep principle, of the fundamental causes for what we readers experience, and not of the surface that everyone recognizes. And to bring people, actual human beings, into the mix! How excellent. How rare. The authors have made this third edition, with its additions, entirely terrific." —Lee Devin and Rob Austin, Co-authors of *The Soul of Design* and *Artful Making* For this third edition, the authors have added six new chapters and updated the text throughout, bringing it in line with today's development environments and challenges. For example, the book now discusses pathologies of leadership that hadn't previously been judged to be pathological; an evolving culture of meetings; hybrid teams made up of people from seemingly incompatible generations; and a growing awareness that some of our most common tools are more like anchors than propellers. Anyone who needs to manage a software project or software organization will find invaluable advice throughout the book.

---

## MICRO FRONTENDS IN ACTION

---

*Manning Publications* **Browser-based software can quickly become complex and difficult to maintain, especially when it's implemented as a large single-page application. By adopting the micro frontends approach and designing your web apps as systems of features, you can deliver faster feature development, easier upgrades, and pick and choose the technology you use in your stack. Micro Frontends in Action is your guide to simplifying unwieldy frontends by composing them from small, well-defined units. Summary Browser-based software can quickly become complex and difficult to maintain, especially when it's implemented as a large single-page application. By adopting the micro frontends approach and designing your web apps as systems of features, you can deliver faster feature development, easier upgrades, and pick and choose the technology you use in your stack. Micro Frontends in Action is your guide to simplifying unwieldy frontends by composing them from small, well-defined units. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology Micro frontends deliver the same flexibility and maintainability to browser-based applications that microservices provide for backend systems. You design your project as a set of standalone components that include their own interfaces, logic, and storage. Then you develop these mini-applications independently and compose them in the browser. About the Book Micro Frontends in Action teaches you to apply the microservices approach to the frontend. You'll start with the core micro frontend design ideas. Then, you'll build an e-commerce application, working through practical issues like server-side and client-side composition, routing, and maintaining a consistent look and feel. Finally, you'll explore team workflow patterns that maximize the benefit of developing application components independently. What's Inside**

- Create a unified frontend from independent applications
- Combine JavaScript code from multiple frameworks
- Browser and server-side composition and routing
- Implement effective dev teams and project workflow

**About the Reader** For web developers, software architects, and team leaders. **About the Author** Michael Geers is a software developer specializing in building user interfaces. **Table of Contents**

**PART 1 - GETTING STARTED WITH MICRO FRONTENDS**

- 1 What are micro frontends?
- 2 My first micro frontends project

**PART 2 - ROUTING, COMPOSITION, AND COMMUNICATION**

- 3 Composition with Ajax and server-side routing
- 4 Server-side composition
- 5 Client-side composition
- 6 Communication patterns
- 7 Client-side routing and the application shell
- 8 Composition and universal rendering

**Which architecture fits my project?**

**PART 3 - HOW TO BE FAST, CONSISTENT, AND EFFECTIVE**

- 10 Asset loading
- 11 Performance is key
- 12 User interface and design system
- 13 Teams and boundaries
- 14 Migration, local development, and testing

---

## JQUERY IN ACTION

---

*Createspace Independent Publishing Platform* **jQuery** was written to express, in short-order, the concepts essential to intermediate and advanced jQuery development. Its purpose is to instill in you, the reader, practices that jQuery developers take as common knowledge. Each chapter contains concepts essential to becoming a seasoned jQuery developer. This book is intended for two types of readers. The first is someone who has read introductory material on jQuery and is looking for the next logical step. The second type of reader is a JavaScript developer, already versed in another library, now trying to quickly learn jQuery. I crafted this book to be used as my own personal reference point for jQuery concepts. This is exactly the type of book I wish every JavaScript library had available. This updated and expanded second edition of Book provides a user-friendly introduction to the subject, Taking a clear structural framework, it guides the reader through the subject's core elements. A flowing writing style combines with the use of illustrations and diagrams throughout the text to ensure the reader understands even the most complex of concepts. This succinct and enlightening overview is a required reading for all those interested in the subject . We hope you find this book useful in shaping your future career & Business.