

[MOBI] Architecting For Scale

Eventually, you will certainly discover a additional experience and achievement by spending more cash. yet when? reach you bow to that you require to get those every needs taking into consideration having significantly cash? Why dont you try to acquire something basic in the beginning? Thats something that will lead you to comprehend even more on the order of the globe, experience, some places, bearing in mind history, amusement, and a lot more?

It is your definitely own epoch to behave reviewing habit. in the course of guides you could enjoy now is **architecting for scale** below.

Architecting for Scale-Lee Atchison 2020-02-28 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

Architecting for Scale-Lee Atchison 2016-07-11 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. This practical guide shows IT, devops, and system reliability managers how to prevent an application from becoming slow, inconsistent, or downright unavailable as it grows. Scaling isn't just about handling more users; it's also about managing risk and ensuring availability. Author Lee Atchison provides basic techniques for building applications that can handle huge quantities of traffic, data, and demand without affecting the quality your customers expect. In five parts, this book explores: Availability: learn techniques for building highly available applications, and for tracking and improving availability going forward Risk management: identify, mitigate, and manage risks in your application, test your recovery/disaster plans, and build out systems that contain fewer risks Services and microservices: understand the value of services for building complicated applications that need to operate at higher scale Scaling applications: assign services to specific teams, label the criticalness of each service, and devise failure scenarios and recovery plans Cloud services: understand the structure of cloud-based services, resource allocation, and service distribution

Architecting for Scale, 2nd Edition-Lee Atchison 2020 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. And 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. Author Lee Atchison shows architects, managers, and directors in both engineering and operations organizations how to build applications at scale that run more smoothly and reliably and meet the needs of your customers. See 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 as your application scales Understand, measure, and mitigate risk in your systems Use the cloud to build highly scalable applications.

Architecting for Scale-Lee Atchison 2016-05-25 As web applications grow, two things begin to happen. They become significantly more complicated and hence brittle, and they handle significantly larger traffic volume requiring more novel and complicated mechanisms to handle this traffic. This can lead to a death spiral for an application that can lead to brownouts, blackouts, and other quality of service and availability problems. This book helps you avoid this death spiral by teaching you basic techniques for building applications that can reliably handle huge quantities of traffic, as well as huge variability in traffic without impacting the quality your customers expect. You'll learn about: Service Oriented Architectures, including services and microservices, for building distributed systems that are more resilient to localized failures Building systems with failures in mind to help you develop robust designs and systems emerge that reduce brittleness and result in higher availability Forward-looking risk management for identifying system problems Scaling your engineering team to build and manage web applications that scale successfully Monitoring and responsiveness to quickly identify problems when your application begins to act abnormally

Architecting Modern Data Platforms-Jan Kunigk 2018-12-05 There's a lot of information about big data technologies, but splicing these technologies into an end-to-end enterprise data platform is a daunting task not widely covered. With this practical book, you'll learn how to build big data infrastructure both on-premises and in the cloud and successfully architect a modern data platform. Ideal for enterprise architects, IT managers, application architects, and data engineers, this book shows you how to overcome the many challenges that emerge during Hadoop projects. You'll explore the vast landscape of tools available in the Hadoop and big data realm in a thorough technical primer before diving into: Infrastructure: Look at all component layers in a modern data platform, from the server to the data center, to establish a solid foundation for data in your enterprise Platform: Understand aspects of deployment, operation, security, high availability, and disaster recovery, along with everything you need to know to integrate your platform with the rest of your enterprise IT Taking Hadoop to the cloud: Learn the important architectural aspects of running a big data platform in the cloud while maintaining enterprise security and high availability

Scale-Gerald Adler 2013-01-11 Scale is a word which underlies much of architectural and urban design practice, its history and theory, and its technology. Its connotations have traditionally been linked with the humanities, in the sense of relating to human societies and to human form. 'To build in scale' is an aspiration that is usually taken for granted by most of those involved in architectural production, as well as by members of the public; yet in a world where value systems of all kinds are being questioned, the term has come under renewed scrutiny. The older, more particular, meanings in the humanities, pertaining to classical Western culture, are where the sense of scale often resides in cultural production. Scale may be traced back, ultimately, to the discovery of musical harmonies, and in the arithmetic proportional relationship of the building to its parts. One might question the continued relevance of this understanding of scale in the global world of today. What, in other words, is culturally specific about scale? And what does scale mean in a world where an intuitive, visual understanding is often undermined or superseded by other senses, or by hyper-reality? Structured thematically in three parts, this book addresses various issues of scale. The book includes an introduction which sets the scene in terms of current architectural discourse and also contains a visual essay in each section. It is of interest to undergraduate and postgraduate students, academics and practitioners in architecture and architectural theory as well as to students in a range of other disciplines including art history and theory, geography, anthropology and landscape architecture.

Large-Scale Software Architecture-Jeff Garland 2003-07-25 The purpose of large-scale software architecture is to capture and describe practical representations to make development teams more effective. In this book the authors show how to utilise software architecture as a tool to guide the development instead of capturing the architectural details after all the design decisions have been made. * Offers a concise description of UML usage for large-scale architecture * Discusses software architecture and design principles * Technology and vendor independent

The Art of Scalability-Martin L. Abbott 2015-05-23 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

ARCHITECTING FOR SCALE.-LEE. ATCHISON 2016

Building Scalable Web Sites-Cal Henderson 2006-05-16 Learn the tricks of the trade so you can build and architect applications that scale quickly--without all the high-priced headaches and service-level agreements associated with enterprise app servers and proprietary programming and database products. Culled from the experience of the Flickr.com lead developer, Building Scalable Web Sites offers techniques for creating fast sites that your visitors will find a pleasure to use. Creating popular sites requires much more than fast hardware with lots of memory and hard drive space. It requires thinking about how to grow over time, how to make the same resources accessible to audiences with different expectations, and how to have a team of developers work on a site without creating new problems for visitors and for each other. Presenting information to visitors from all over the world Integrating email with your web applications Planning hardware purchases and hosting options to have as much as you need without breaking your wallet Partitioning and distributing databases to support large datasets and simultaneous transactions Monitoring your applications to find and clear bottlenecks * Providing services APIs and using services from other providers to increase your site's reach and capabilities Whether you're starting a small web site with hopes of growing big or you already have a large system that needs maintenance, you'll find Building Scalable Web Sites to be a library of ideas for making things work.

Designing Data-Intensive Applications-Martin Kleppmann 2017-03-16 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

Architecting Data-Intensive Applications-Anuj Kumar 2018-07-31 Architect and design data-intensive applications and, in the process, learn how to collect, process, store, govern, and expose data for a variety of use cases Key Features Integrate the data-intensive approach into your application architecture Create a robust application layout with effective messaging and data querying architecture Enable smooth data flow and make the data of your application intensive and fast Book Description Are you an architect or a developer who looks at your own applications gingerly while browsing through Facebook and applauding it silently for its data-intensive, yet fluent and efficient, behaviour? This book is your gateway to build smart data-intensive systems by incorporating the core data-intensive architectural principles, patterns, and techniques directly into your application architecture. This book starts by taking you through the primary design challenges involved with architecting data-intensive applications. You will learn how to implement datauration and data dissemination, depending on the volume of your data. You will then implement your application architecture one step at a time. You will get to grips with implementing the correct message delivery protocols and creating a data layer that doesn't fail when running high traffic. This book will show you how you can divide your application into layers, each of which adheres to the single responsibility principle. By the end of this book, you will learn to streamline your thoughts and make the right choice in terms of technologies and architectural principles based on the problem at hand. What you will learn Understand how to envision a data-intensive system Identify and compare the non-functional requirements of a data collection component Understand patterns involving data processing, as well as technologies that help to speed up the development of data processing systems Understand how to implement Data Governance policies at design time using various Open Source Tools Recognize the anti-patterns to avoid while designing a data store for applications Understand the different data dissemination technologies available to query the data in an efficient manner Implement a simple data governance policy that can be extended using Apache Falcon Who this book is for This book is for developers and data architects who have to code, test, deploy, and/or maintain large-scale, high data volume applications. It is also useful for system architects who need to understand various non-functional aspects revolving around Data Intensive Systems.

Architecting High Performing, Scalable and Available Enterprise Web Applications-Shailesh Kumar Shivakumar 2014-11-11 Provides in-depth insights into techniques for achieving desired scalability, availability and performance quality goals for enterprise web applications. The book provides an integrated 360-degree view of achieving and maintaining these attributes through practical, proven patterns, novel models, best practices, performance strategies, and continuous improvement methodologies and case studies.

Architecting with RM-ODP-Janis Putman 2001 PrefaceTo understand anything, you should not try to understand everything. — Aristotle The whole is greater than the sum of the parts; the part is greater than a fraction of the whole. — Aristotle Architecting is a challenging process of abstraction, composition, modularity, and simplification to create an architecture specification. An architecture specification captures the essence and definition of the system: understanding, parts, and the relationships among the parts. An architecture specification defines how a system solves a business problem within the scope of the business. — Putman Leave the beaten track occasionally and dive into the woods. You will be certain to find something that you have never seen before. — Alexander Graham Bell There are large gaps in the theory and practice of software architecture and engineering. Much is published about the representation of a software architecture, such as the Unified Modeling Language (UML), but little is available about the specification for a software architecture. Software engineering methods of domain engineering, process modeling languages, and well-formed patterns of reasoning aid in the specification of an architecture. The Reference Model of Open Distributed Processing (RM-ODP) defines the standard reference model for distributed software systems architectures, based on object-oriented techniques, accepted at the international level. RM-ODP is a standard adopted by the International Standards Organization (ISO) and the International Telecommunications Union (ITU). RM-ODP is embedded and used actively in mission-critical systems industries such as in telecommunications, in health care, on Wall Street (financial services industry), in various Government systems (Logistics), in European Government Agencies such as UK Aviation control systems, as a foundation for the Object Management Group (OMG) Object Management Architecture (OMA), for defining enterprise architectures, and for defining software architectures. The software systems architecture work that is emerging, and is focused either at the component level or at the systems level, provides a key resource for architecting. This is enhanced by the architecting techniques of RM-ODP. This book assembles these great ideas, explains what they mean, and shows how to use them for practical benefit, along with real-world case study examples. By using the RM-ODP specification constructs, associated languages, architecture patterns of reasoning, semantic behavior specification, and conformance testing abilities, readers will be able to architect their specific systems based on the RM-ODP specification foundations, and specify architectures that work. One of the purposes of this book is to provide the approach to using the RM-ODP foundations in architecting and specifying a distributed processing system that addresses such key properties as interoperability, dependability, portability, integration, composability, scalability, transparency, behavior specification, quality of service, policy management, federation, and conformance validation. Another purpose of this book is to explain the underlying foundations for creating an architectural specification. These foundations come not only from RM-ODP, but also from the current work in software systems architecture. Another purpose is to guide the reader to understand the importance and benefits of creating an architecture specification for an enterprise. Yet another

purpose is to provide the reader with the principles to construct software systems architecture (at both introductory and in-depth levels). By applying the proven techniques of RM-ODP for what makes a good architecture, readers will be able to build their own tailored architectures, and clearly represent them in UML or some other tool, with an understanding of the underlying principles. Practitioners of RM-ODP have found that the standard is extremely beneficial in guiding architecture definition and providing standard terminology/principles for distributed object applications and infrastructures from an enterprise perspective. Outstanding Features This book is intended to provide valuable insight into successful architecture specification by describing an unprecedented foundation to accomplish this task, describing the use of the foundation, explaining the relationships of the concepts of architecting, explaining the relationships of the concepts of distributed processing, and identifying the right methods and possible tools for architecting. All material for the book has been derived from actual experiences. A medical case study is used throughout the book in ever increasing detailed specification. This medical case study is based on actual experience of the author. In addition, many metamodels are provided to represent the concepts of RM-ODP. All of these metamodels are contributions from the author. This is information that readers can use and apply in their architecting today. RM-ODP provides a reference framework, grammars, methods of abstraction and composition, and separation of concerns to achieve an architecture specification of the system. RM-ODP provides a framework for this separation, using viewpoints, as well as separating out certain decisions (e.g., product decisions) until later. Further, the reference model provides a set of definitions, which always aids in communicating with others. There is little in the literature about RM-ODP or architecture specification, and certainly not a book dedicated as a tutorial of these subjects. Now there is. In summary, this book offers the following: How to manage the architecting process in the lifecycle of a system How to solve many architecture reuse and cost-effectiveness problems How to create a business specification How to understand and use the concepts of distributed processing in an architecture How to architect effectively How to specify an architecture How to understand and specify semantic behavior and nonfunctional properties of a system (the "ilities") How to provide the right level of detail in an architecture specification How to ensure the implementation conforms to the architecture specification How to use RM-ODP effectively How to use popular tools, such as UML, to describe an architecture A definitive tutorial of RM-ODP Audience This book is designed for: Those in the Distributed Software Systems Architecture community who are interested in a methodology for using proven architecture principles. Professional software architects who are looking for new ideas about architecting a system. Within this book, the reader will find discussions of the techniques for architecting, for creating an architecture specification, and RM-ODP's relationship to other architecture frameworks. Program managers interested in how to create a cost-effective architecture within their enterprise that focuses on the needs of the enterprise and solves an enterprise problem. They will learn how to do this through an overview of RM-ODP, the program benefits for using it, and where RM-ODP fits in the system lifecycle process. Systems engineers interested in the lifecycle approach to enterprise architecture specification. Experienced engineers interested in expanding their understanding of how to create a valid architecture specification and gain an understanding of the distributed processing system concepts, why certain constructions are valid and why some are not, what is to be specified and how, and some new ideas and approaches to architecting a system. The reader will be able to develop a collection of useful distributed processing architecting techniques that expand upon the current software systems architecture capabilities. Developers interested in the practice of architecture specification and aligning current technology to achieve a workable system, while allowing evolutionary changes in technology solutions. Researchers interested in solutions and aids for furthering the research work in architecture specification. Individuals in the software community who are generally interested in the application of an architecture method. Readers will find examples of the applications of RM-ODP and specific analysis techniques. The expected audience will be novice and mid-level program managers, software engineers, those in the IEEE, DoD, research communities, consortia, and general architecture readers. This book can be used as a textbook and reference book for studies in the methods of architecture; for graduate studies in software architecture specification; for training information about software architecture and RM-ODP; for further education of consultants, integration specialists, and acquisition managers who need to approve and fund such work; and for researchers who are expanding the discipline of software architecture. The inclusion of RM-ODP will bring to the U.S., principally, the outstanding work that was accomplished by the international standards working group. In brief, the RM-ODP principles form a solution set and foundation for all software architecting endeavors. It is the formalized framework for this topic, and at the International Standard (IS) level of acceptance. It forms a solution set and foundation for reuse of design patterns to provide cost-effective software architecture. It is the process for this topic, but has never before been described in a book. Many program managers (who typically set the stage as to the methodology of choice for a project), software engineers, and researchers in academia and in DARPA are unaware of the power and solutions provided by the standard, or the process of identifying and instantiating reuse of all the expensive assets of architecture. Many do not realize that there is a language for specifying software-intensive distributed processing, and that language is precisely and rigorously defined in RM-ODP for reuse. Those debating definitions for architecture, system, interface, and others can reuse the internationally agreed upon definitions. Finally, with the inclusion of RM-ODP and its relationship to other architecture frameworks, it is expected that many software engineers will benefit from reading this work, since it will be the first time these subjects are discussed in print. How to Use This Book This book is divided into four parts, aimed at increasing levels of detail. Part One provides an overview of the field of software architecture, an RM-ODP primer for managers, and an RM-ODP primer for architects. Part Two provides an in-depth study of RM-ODP and how to use it. Areas of importance and utility from RM-ODP are highlighted. Ambiguity in RM-ODP is highlighted. Warnings in the use of RM-ODP are highlighted. Part Three provides a discussion of the principal architecture patterns of use, arranged by topic. Several of these patterns of use come from emerging work under the initiative of RM-ODP, as well as lessons learned from the practice of RM-ODP. These patterns of reasoning used by the architect are founded on the principals of RM-ODP, as discussed in Part Two of the book. Part Four concludes with relating RM-ODP to other architecture methods. It also provides emerging technologies to further the patterns of reasoning for use in architecting, and a set of architecting heuristics. The information contained in this book is organized in a manner that provides clear insight into the world of distributed software-intensive processing architecture for designers and developers who are familiar with information systems technology, but want to know more about how to build a good architecture. Starting with a tutorial about software architecture, and then a tutorial about the standard for software architecture, the reader need not be an expert in the area of international standards, RM-ODP, software architecture, or specific technologies. The book goes on to address the needs of the variety of readers for which it is intended. Each chapter in the book provides an overview of the subject of the chapter, as well as a summary. For those who wish a broad brush exposure to RM-ODP, the primers of Part One provide this, as well as the overviews and summaries in each chapter of interest. As each chapter progresses, in Parts Two and Three, more and more in-depth detail is provided. The readings of these chapters are aimed at those who wish to know the technical details of a topic. There are two case studies used throughout the book, at various levels of detail. The primary case study is a Hospital enterprise, based upon the author's experience with the medical profession. A secondary case study is an airline reservation system, also based upon the author's experience. These case studies are used to describe the concepts of RM-ODP, and to show how they might be used.

Architecting Complex-Event Processing Solutions with TIBCO®-Paul C. Brown 2013-09-20 "Complex-event processing is simple in principle but hard to do well in practice. This guide presents the principles and motivations for those new to the subject. More importantly, it details the entire thought-landscape of a complete implementation, using TIBCO products as the background. Well worth the read for anyone who is thinking of implementing a complex-event solution. Those who have already implemented one should read it as well, both for another perspective and for a view of the capabilities of the TIBCO products." —Lloyd Fischer, Senior Software Architect, WellCare Health Plans "This complete guide drives you through the specifics of complex-event processing (CEP) design concepts. The book covers all the fundamental aspects and design phases relevant for any TIBCO CEP project implementation, from design through performance-tuning and deployment. I would highly recommend this book to any reader interested in CEP concepts, although a small amount of TIBCO technology knowledge will let you appreciate it more." —Antonio Bruno, Infrastructure Account Manager, UBS AG The architecture series from TIBCO® Press comprises a coordinated set of titles for software architects and developers, showing how to combine TIBCO components to design and build real-world solutions. Complex-event processing is required when multiple events occurring throughout an organization must be sensed, analyzed, prioritized, and acted on in real time. Architecting Complex-Event Processing Solutions with TIBCO® shows how to design and architect complex-event processing systems, addressing all their complexities and achieving maximum efficiency and effectiveness, while delivering superior business value. After reading this book, you will be able to Identify opportunities for competitive differentiation through complex-event processing Describe differences between complex-event processing and traditional systems Understand relevant capabilities of the TIBCO BusinessEvents™ product suite Select building-block design patterns for constructing complex-event processing solutions with TIBCO BusinessEvents Address architectural aspects of moving solutions into production Implement proven approaches to designing fault tolerance and high availability Architecting Complex-Event Processing Solutions with TIBCO® is intended for working architects, designers, and developers who want to apply TIBCO products in complex-event processing applications. It is also required reading for anyone seeking TIBCO Certified Architect status.

The Hacker's Guide to Scaling Python-Julien Danjou Python is a wonderful programming language that allows writing applications quickly. But how do you make those applications scale for thousands of users and requests? It takes years of practice, research, trial and errors to build experience and knowledge along the way. Simple questions such as "How do I make my code faster?" or "How do I make sure there is no bottleneck?" cost hours to find good answers. Without enough background on the topic, you'll never be sure that any answer you'll come up with will be correct. The Hacker's Guide to Scaling Python will help you solve that by providing guidelines, tips

and best practice. Adding a few interviews of experts on the subject, you will learn how you can distribute your Python application so it is able to process thousands of requests.

Systems Engineering and Architecting-Laurence Bellagamba 2012-03-26 Systems Engineering and Architecting: Creating Formal Requirements presents formal requirements to help you accomplish key systems engineering and architecting activities more efficiently. The formal requirements—explicit, executable, verifiable instructions—explain how to model systems behavior, make decisions, establish natural language requirements, and improve your systems engineering and architecting processes. Each chapter opens with case studies and lessons learned, which supply the real-world context for the formal requirements. Topics covered include how to use fuzzy logic and agents to model uncertainty and how to make decisions when confronted with ambiguity. The book also clarifies the differences between architecting and systems engineering. Mathematical Tools for Systems Engineering and Architecting Written in Mathematica®, each formal requirement provides a tool or serves as the algorithm for a more efficient implementation in another form. All of the requirements are available as an open source library for anyone to use, improve upon, or add to. Worked examples, illustrations, and example surveys help you apply the requirements to your own systems. The book also lists heuristics to guide you in those systems engineering or architecting activities that cannot yet be formally stipulated. Bring More Consistency to Your Systems Development and Management Acknowledging that much of the practice remains an art, this book brings as much scientific rigor as possible to the tasks performed by systems engineers and architects. Written by a director of engineering who led systems engineering or architecting efforts for the Space Shuttle Program, Space Control Architecture Development, and others, this book shows you how to develop more consistent processes for large-scale systems.

Scaling Your Node.js Apps-Fernando Doglio 2018-11-28 Take your Node.js application into production-ready status, capable of scaling up to whatever your needs might be. You'll discover that architecting for successful, popular sites is an essential tool of any professional Node.js developer, and learning to scale your own applications is a great place to start. Using this book you will learn when to scale, what factors should trigger scaling, and what architectural techniques are best suited for scaling. You will also explore common pitfalls that arise when scaling a Node.js application and solutions to correct them. Including analyses of success cases at the largest-scale companies, such as Netflix and Paypal, this book will get you started with scaling in no time at all. What You'll Learn Determine what factors should trigger the need to scale Discover different architectural patterns that lend themselves to scaling Resolve problems that arise when scaling up a Node.js application Monitor a platform in order to understand when to start scaling Who This Book Is For The main audience for this book are Node.js developers with a mid-level understanding of the technology. Novice Node users will also benefit from the coverage of generic scaling-related topics.

AWS Certified Solutions Architect Official Study Guide-Joe Baron 2016-10-17 Validate your AWS skills. This is your opportunity to take the next step in your career by expanding and validating your skills on the AWS cloud. AWS has been the frontrunner in cloud computing products and services, and the AWS Certified Solutions Architect Official Study Guide for the Associate exam will get you fully prepared through expert content, and real-world knowledge, key exam essentials, chapter review questions, access to Sybex's interactive online learning environment, and much more. This official study guide, written by AWS experts, covers exam concepts, and provides key review on exam topics, including: Mapping Multi-Tier Architectures to AWS Services, such as web/app servers, firewalls, caches and load balancers Understanding managed RDBMS through AWS RDS (MySQL, Oracle, SQL Server, Postgres, Aurora) Understanding Loose Coupling and Stateless Systems Comparing Different Consistency Models in AWS Services Understanding how AWS CloudFront can make your application more cost efficient, faster and secure Implementing Route tables, Access Control Lists, Firewalls, NAT, and DNS Applying AWS Security Features along with traditional Information and Application Security Using Compute, Networking, Storage, and Database AWS services Architecting Large Scale Distributed Systems Understanding of Elasticity and Scalability Concepts Understanding of Network Technologies Relating to AWS Deploying and Managing Services with tools such as CloudFormation, OpsWorks and Elastic Beanstalk. Learn from the AWS subject-matter experts, review with proven study tools, and apply real-world scenarios. If you are looking to take the AWS Certified Solutions Architect Associate exam, this guide is what you need for comprehensive content and robust study tools that will help you gain the edge on exam day and throughout your career.

Architecting Cloud Native Applications-Kamal Arora 2019-04-16 Apply cloud native patterns and practices to deliver responsive, resilient, elastic, and message-driven systems with confidence Key Features Discover best practices for applying cloud native patterns to your cloud applications Explore ways to effectively plan resources and technology stacks for high security and fault tolerance Gain insight into core architectural principles using real-world examples Book Description Cloud computing has proven to be the most revolutionary IT development since virtualization. Cloud native architectures give you the benefit of more flexibility over legacy systems. This Learning Path teaches you everything you need to know for designing industry-grade cloud applications and efficiently migrating your business to the cloud. It begins by exploring the basic patterns that turn your database inside out to achieve massive scalability. You'll learn how to develop cloud native architectures using microservices and serverless computing as your design principles. Then, you'll explore ways to continuously deliver production code by implementing continuous observability in production. In the concluding chapters, you'll learn about various public cloud architectures ranging from AWS and Azure to the Google Cloud Platform, and understand the future trends and expectations of cloud providers. By the end of this Learning Path, you'll have learned the techniques to adopt cloud native architectures that meet your business requirements. This Learning Path includes content from the following Packt products: Cloud Native Development Patterns and Best Practices by John Gilbert Cloud Native Architectures by Erik Farr et al. What you will learn Understand the difference between cloud native and traditional architecture Automate security controls and configuration management Minimize risk by evolving your monolithic systems into cloud native applications Explore the aspects of migration, when and why to use it Apply modern delivery and testing methods to continuously deliver production code Enable massive scaling by turning your database inside out Who this book is for This Learning Path is designed for developers who want to progress into building cloud native systems and are keen to learn the patterns involved. Software architects, who are keen on designing scalable and highly available cloud native applications, will also find this Learning Path very useful. To easily grasp these concepts, you will need basic knowledge of programming and cloud computing.

Architecting Modern Java EE Applications-Sebastian Daschner 2017-10-09 Find out how to craft effective, business-oriented Java EE 8 applications that target customer's demands in the age of Cloud platforms and container technology. About This Book Understand the principles of modern Java EE and how to realize effective architectures Gain knowledge of how to design enterprise software in the age of automation, Continuous Delivery and Cloud platforms Learn about the reasoning and motivations behind state-of-the-art enterprise Java technology, that focuses on business Who This Book Is For This book is for experienced Java EE developers who are aspiring to become the architects of enterprise-grade applications, or software architects who would like to leverage Java EE to create effective blueprints of applications. What You Will Learn What enterprise software engineers should focus on Implement applications, packages, and components in a modern way Design and structure application architectures Discover how to realize technical and cross-cutting aspects Get to grips with containers and container orchestration technology Realize zero-dependency, 12-factor, and Cloud-native applications Implement automated, fast, reliable, and maintainable software tests Discover distributed system architectures and their requirements In Detail Java EE 8 brings with it a load of features, mainly targeting newer architectures such as microservices, modernized security APIs, and cloud deployments. This book will teach you to design and develop modern, business-oriented applications using Java EE 8. It shows how to structure systems and applications, and how design patterns and Domain Driven Design aspects are realized in the age of Java EE 8. You will learn about the concepts and principles behind Java EE applications, and how to effect communication, persistence, technical and cross-cutting concerns, and asynchronous behavior. This book covers Continuous Delivery, DevOps, infrastructure-as-code, containers, container orchestration technologies, such as Docker and Kubernetes, and why and especially how Java EE fits into this world. It also covers the requirements behind containerized, zero-dependency applications and how modern Java EE application servers support these approaches. You will also learn about automated, fast, and reliable software tests, in different test levels, scopes, and test technologies. This book covers the prerequisites and challenges of distributed systems that lead to microservice, shared-nothing architectures. The challenges and solutions of consistency versus scalability will further lead us to event sourcing, event-driven architectures, and the CQRS principle. This book also includes the nuts and bolts of application performance as well as how to realize resilience, logging, monitoring and tracing in a modern enterprise world. Last but not least the demands of securing enterprise systems are covered. By the end, you will understand the ins and outs of Java EE so that you can make critical design decisions that not only live up to, but also surpass your clients' expectations. Style and approach This book focuses on solving business problems and meeting customer demands in the enterprise world. It covers how to create enterprise applications with reasonable technology choices, free of cargo-cult and over-engineering. The aspects shown in this book not only demonstrate how to realize a certain solution, but also explain its motivations and reasoning.

Production-Ready Microservices-Susan J. Fowler 2016-11-30 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

Programming AWS Lambda-John Chapin 2020-03-18 Serverless revolutionizes the way organizations build and deploy software. With this hands-on guide, Java engineers will learn how to use their experience in the new world of serverless computing. You'll discover how this cloud computing execution model can drastically decrease the complexity in developing and operating applications while reducing costs and time to market. Engineering leaders John Chapin and Mike Roberts guide you through the process of developing these applications using AWS Lambda, Amazon's event-driven, serverless computing platform. You'll learn how to prepare the development environment, program Lambda functions, and deploy and operate your serverless software. The chapters include exercises to help you through each aspect of the process. Get an introduction to serverless, functions as a service, and AWS Lambda Learn how to deploy working Lambda functions to the cloud Program Lambda functions and learn how the Lambda platform integrates with other AWS services Build and package Java-based Lambda code and dependencies Create serverless applications by building a serverless API and data pipeline Test your serverless applications using automated techniques Apply advanced techniques to build production-ready applications Understand both the gotchas and new opportunities of serverless architecture

Information Architecture for the World Wide Web-Louis Rosenfeld 2002 Discusses Web site hierarchy, usability, navigation systems, content labeling, configuring search systems, and managing the information architecture development process.

Cloud Architecture Patterns-Bill Wilder 2012 Do you need to learn about cloud computing architecture with Microsoft's Azure quickly? Read this book! It gives you just enough info on the big picture and is filled with key terminology so that you can join the discussion on cloud architecture.

Architecting HBase Applications-Jean-Marc Spaggiari 2016-07-18 HBase is a remarkable tool for indexing mass volumes of data, but getting started with this distributed database and its ecosystem can be daunting. With this hands-on guide, you'll learn how to architect, design, and deploy your own HBase applications by examining real-world solutions. Along with HBase principles and cluster deployment guidelines, this book includes in-depth case studies that demonstrate how large companies solved specific use cases with HBase. Authors Jean-Marc Spaggiari and Kevin O'Dell also provide draft solutions and code examples to help you implement your own versions of those use cases, from master data management (MDM) and document storage to near real-time event processing. You'll also learn troubleshooting techniques to help you avoid common deployment mistakes. Learn exactly what HBase does, what its ecosystem includes, and how to set up your environment Explore how real-world HBase instances were deployed and put into production Examine documented use cases for tracking healthcare claims, digital advertising, data management, and product quality Understand how HBase works with tools and techniques such as Spark, Kafka, MapReduce, and the Java API Learn how to identify the causes and understand the consequences of the most common HBase issues

Web Application Security-Andrew Hoffman 2020-03-02 While many resources for network and IT security are available, detailed knowledge regarding modern web application security has been lacking—until now. This practical guide provides both offensive and defensive security concepts that software engineers can easily learn and apply. Andrew Hoffman, a senior security engineer at Salesforce, introduces three pillars of web application security: recon, offense, and defense. You'll learn methods for effectively researching and analyzing modern web applications—including those you don't have direct access to. You'll also learn how to break into web applications using the latest hacking techniques. Finally, you'll learn how to develop mitigations for use in your own web applications to protect against hackers. Explore common vulnerabilities plaguing today's web applications Learn essential hacking techniques attackers use to exploit applications Map and document web applications for which you don't have direct access Develop and deploy customized exploits that can bypass common defenses Develop and deploy mitigations to protect your applications against hackers Integrate secure coding best practices into your development lifecycle Get practical tips to help you improve the overall security of your web applications

The Art of Systems Architecting-Eberhardt Rechtin 2010-12-12 Today's architecting must handle systems of types unknown until very recently. New domains, including personal computers, intersatellite networks, health services, and joint service command and control are calling for new architectures-and for architects specializing in those domains. Since the original publication, of this bestselling text, these

Architecting Robust Co-Design of Materials, Products, and Manufacturing Processes-Anand Balu Nellippallil 2020-06-13 This book explores systems-based, co-design, introducing a "Decision-Based, Co-Design" (DBCD) approach for the co-design of materials, products, and processes. In recent years there have been significant advances in modeling and simulation of material behavior, from the smallest atomic scale to the macro scale. However, the uncertainties associated with these approaches and models across different scales need to be addressed to enable decision-making resulting in designs that are robust, that is, relatively insensitive to uncertainties. An approach that facilitates co-design is needed across material, product design and manufacturing processes. This book describes a cloud-based platform to support decisions in the design of engineered systems (CB-PDSIDES), which feature an architecture that promotes co-design through the servitization of decision-making, knowledge capture and use templates that allow previous solutions to be reused. Placing the platform in the cloud aids mass collaboration and open innovation. A valuable reference resource reference on all areas related to the design of materials, products and processes, the book appeals to material scientists, design engineers and all those involved in the emerging interdisciplinary field of integrated computational materials engineering (ICME).

Architecting Systems-Hillary Sillitto 2014

Building Secure and Reliable Systems-Heather Adkins 2020-03-16 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

Millennials and the Workplace-Pritam Singh 2012-06-13 Millennials and the Workplace explores the rapidly changing workplace environment and its meaning in the new millennium. It takes a look at the young generation—the Millennials—who do not accept the 'status quo', who have moved from obedience to questioning, and who assess work in terms of its significance in human life and human nature. The book studies the Millennials and profiles them according to their demographic and psychographic background along with personality factors such as work locus of control, personal values and achievement motivation, and so on. The authors highlight the gap between what the Millennials expect and what they get at the workplace. The contours of the new workplace—creating alignment between the individual's expectations and the organization's offering—have been sketched in detail for retaining talent. With these insights, the book will be immensely useful to all corporate professionals, academia, and students.

Software Engineering at Google-Titus Winters 2020-02-28 Today, software engineers need to know not only how to program effectively but also how to develop proper engineering practices to make their codebase sustainable and healthy. This book emphasizes this difference between programming and software engineering. How can software engineers manage a living codebase that evolves and responds to changing requirements and demands over the length of its life? Based on their experience at Google, software engineers Titus Winters and Hyrum Wright, along with technical writer Tom Manshreck, present a candid and insightful look at how some of the world's leading practitioners construct and maintain software. This book covers Google's unique engineering culture, processes, and tools and how these aspects contribute to the effectiveness of an engineering organization. You'll explore three fundamental principles that software organizations should keep in mind when designing, architecting, writing, and maintaining code: How time affects the sustainability of software and how to make your code resilient over time How scale affects the viability of software practices within an engineering organization What trade-offs a typical engineer needs to make when evaluating design and development decisions

Enterprise Rails-Dan Chak 2008-10-21 What does it take to develop an enterprise application with Rails? Enterprise Rails introduces several time-tested software engineering principles to prepare you for the challenge of building a high-performance, scalable website with global reach. You'll learn how to design a solid architecture that ties the many parts of an enterprise website together, including the database, your servers and clients, and other services as well. Many Rails developers think that planning for scale is unnecessary. But there's nothing worse than an application that fails because it can't handle sudden success. Throughout this book, you'll work on an example enterprise project to learn first-hand what's involved in architecting serious web applications. With this book, you will: Tour an ideal enterprise systems layout: how Rails fits in, and which elements don't rely on Rails Learn to structure a Rails 2.0 application for complex websites Discover how plugins can support reusable code and improve application clarity Build a solid data model -- a fortress -- that protects your data from corruption Base an ActiveRecord model on a database view, and build support for multiple table inheritance Explore service-oriented architecture and web services with XML-RPC and REST See how caching can be a dependable way to improve performance Building for scale requires more work up front, but you'll have a flexible website that can be extended easily when your needs change. Enterprise Rails teaches you how to architect scalable Rails applications from the ground up. "Enterprise Rails is indispensable for anyone planning to build enterprise web services. It's one thing to get your service off the ground with a framework like Rails, but quite another to construct a system that will hold up at enterprise scale. The secret is to make good architectural choices from the beginning. Chak shows you how to make those choices. Ignore his advice at your peril."-- Hal Abelson, Prof. of Computer Science and Engineering, MIT

Building Evolutionary Architectures-Neal Ford 2017-09-18 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.

The Process of Software Architecting-Peter Eeles 2010 The definitive guide to building superior software architectures: process, roles, artifacts, activities, and much more * *Applicable across ALL technologies and platforms, including both J2EE and .NET. *Includes interviews with many of the world's leading software architects, and a complete start-to-finish case study demonstrating this book's processes at work. *Reviewed and verified by renowned industry leaders including Grady Booch, Roger Blake, and Philippe Kruchten. Software architecture provides the foundation for any software system. To build an effective software architecture, professionals must understand the processes they should be following, the organizational roles they will interact with, the artifacts they must produce, the activities they will perform, and the sequence in which those activities must be performed. This book doesn't just walk through all of these issues in unprecedented depth: it brings together the knowledge and expertise of ten of the world's leading software architects. The authors define a coherent framework for planning and producing architecture in any environment. They cover the entire project lifecycle, introducing best-practice techniques that will be equally applicable in all environments: Java/J2EE, .NET, or otherwise. Readers will find deeply insightful interviews with leading practitioners like Philippe Kruchten (architect of the Canadian Air Traffic Control System) and Roger Blake (the IBM architect who implemented the real-time score-reporting system at the Wimbledon tennis tournament). The authors also present a single case study that demonstrates their processes at work from start to finish.

Multidisciplinary Systems Engineering-James A. Crowder 2015-12-23 This book presents Systems Engineering from a modern, multidisciplinary engineering approach, providing the understanding that all aspects of systems design, systems, software, test, security, maintenance and the full life-cycle must be factored in to any large-scale system design; up front, not factored in later. It lays out a step-by-step approach to systems-of-systems architectural design, describing in detail the documentation flow throughout the systems engineering design process. It provides a straightforward look and the entire systems engineering process, providing realistic case studies, examples, and design problems that will enable students to gain a firm grasp on the fundamentals of modern systems engineering. Included is a comprehensive design problem that weaves throughout the entire text book, concluding with a complete top-level systems architecture for a real-world design problem.

Digital Transformation at Scale-GREENWAY 2018-04-30 "This book is for people worrying about their sinking ship. Based on experience, it is a guide for navigating the blockers, buzzwords and bloody-mindedness that doom any analogue organisation trapped into thinking that while the internet has changed the world, it won't change their world. Companies that grew up on the web have changed our expectations of the services we rely on. We demand simplicity, speed and low cost. Organizations founded before the Internet aren't keeping up - despite spending millions on IT, marketing and 'innovation'. This book is a guide to building a digital institution. It explains how a growing band of reformers in businesses and governments around the world have helped their organizations pivot to this new way of working, and what lessons others can learn from their experience. It is based on the authors' experience designing and helping to deliver the UK's Government Digital Service (GDS). The GDS was a new institution made responsible for the digital transformation of government, designing public services for the Internet era. It snipped £4 billion off the government's technology bill, opened up public sector contracts to thousands of new suppliers, and delivered online services so good that citizens chose to use them over the offline alternatives, without a big marketing campaign. Other countries and companies noticed, with the GDS model now being copied around the world."

The World As an Architectural Project-Hashim Sarkis 2020-02 Architects imagine the planet: fifty speculative world-scale projects from Patrick Geddes, Alison and Peter Smithson, Kiyonori Kikutake, Juan Navarro Baldeweg, Luc Deleu, and others. The world's growing vulnerability to planet-sized risks invites action on a global scale. The World as an Architectural Project shows how for more than a century architects have imagined the future of the planet through world-scale projects. With fifty speculative projects by Patrick Geddes, Alison and Peter Smithson, Kiyonori Kikutake, Saverio Muratori, Takis Zenetos, Sergio Bernades, Juan Navarro Baldeweg, Luc Deleu, and many others, documented in text and images, this ambitious and wide-ranging book is the first compilation of its kind. Interestingly, architects begin to address the world as a project long before the advent of contemporary globalism and its assorted anxieties. The Spanish urban theorist and entrepreneur Arturo Soria y Mata, for example, in 1882 envisions a system that connects the entire planet in a linear urban network. In 1927, Buckminster Fuller's "World Town Plan--4D Tower" proposes to solve global housing problems with mobile structures delivered and installed by a Zeppelin. And Joyce Hsiang and Bimal Mendis visualize the conditions of a worldwide "City of Seven Billion" in a 2015-2019 project. Rather than indulging the cliché of the megalomaniac architect, this volume presents a discipline reflecting on its own responsibilities.

Architecting and Operating OpenShift Clusters-William Caban 2019-12-10 Design and architect resilient OpenShift clusters and gain a keen understanding of how hundreds of projects are integrated into a powerful solution. While there are many OpenShift resources available for developers, this book focuses on the key elements of infrastructure and operations that teams need when looking to integrate and maintain this platform. You'll review important concepts, such as repeatable deployment techniques, advanced OpenShift RBAC capabilities, monitoring clusters, and integrating with external services. You'll also see how to run specialized workloads in OpenShift and how to deploy non-web based applications on the platform, all designed to help cultivate best practices as your organization continue evolve in microservices architectures. OpenShift has become the main enterprise Kubernetes distribution and its market penetration continues to growth at rapid rate. While OpenShift's documentation provides a great list of configuration options to work with the platform, it can be a daunting task to wade through. Architecting and Operating OpenShift Clusters breaks this content down into clear and useful concepts to provide you with a solid understanding of the OpenShift internal architecture. What You'll Learn Operate high availability in multi-tenant OCP clusters Understand OpenShift SDN models, capabilities, and storage classes Integrate OCP with existing data center capabilities and CI/CD pipelines Support

advanced capabilities like: Istio, Multus, Kubernetes Operators, hybrid deployments Who This Book Is For Cloud architects, OpenShift cluster administrators, and teams supporting developers in OpenShift environments who have a basic understanding of this platform and microservices architectures.