Qpid C Programming Guide

Eventually, you will no question discover a additional experience and finishing by spending more cash. nevertheless when? do you bow to that you require to acquire those all needs past having significantly cash? Why dont you try to get something basic in the beginning? Thats something that will lead you to comprehend even more regarding the globe, experience, some places, in the manner of history, amusement, and a lot more?

It is your enormously own get older to be in reviewing habit. along with guides you could enjoy now is **Qpid C Programming Guide** below.

The Next Generation Crew Book IBM
As cloud technology continues to advance and be utilized, many service providers have begun to employ multiple networks, or cloud federations; however, as the popularity of these federations increases, so does potential utilization challenges. Developing Interoperable and Federated Cloud Architecture provides valuable insight into current and emergent research

occurring within the field of cloud infrastructures. Featuring barriers, recent developments, and practical applications on the interoperability issues of federated cloud architectures, this book is a focused reference for administrators, developers, and cloud users interested in energy awareness, scheduling, and federation policies and usage.

Microservices IOS Press

This book analyzes the application of the legal principle of non-discrimination in the context of energy network operation. Since the early 1990s, the duty not to discriminate has applied to energy network operators, in order to achieve a liberalized European energy market in which European consumers have a free and real choice of energy supplier. The book provides guidance to those working in the context of the non-discrimination obligation, such as energy network operators, regulatory authorities, national courts, and other energy market players, as well as those studying the rules for (academic) research purposes. The book's conclusions serve as a tool for critical consideration and offer suggestions for improvements to the legal framework and its application on a European, as well as a national, level. Several questions are answered, including why energy network operators have a non-discrimination obligation in the context of energy

market liberalization, how European law has tried to remove and control the discrimination problem since the early 1990s, and when different treatment of energy network users amounts to 'forbidden' discrimination. The book's conclusions are underpinned by comparisons with competition law, public procurement law, and telecommunications law, as well as a case study on how energy network operators and regulators in several Member States currently interpret and apply the non-discrimination obligation. (Series: Energy & Law - Vol. 15)

97 Things Every Java Programmer Should Know Packt Publishing Ltd

Teaches you how and what to study in order to be best prepared for the Certified OpenStack Administrator exam. This fast-growing technology is creating a market that needs more qualified IT specialists with proven skills. This book covers 100% of the exam requirements for both The OpenStack Foundation and the Mirantis OpenStack Certification Exam. Each theme is taught using practical exercises and instructions for the command line and for the graphical client (Horizon). Each chapter is followed by review questions, complete with answers. Even after you have taken and passed your OpenStack exam, this book will remain a useful reference. What You Will Learn Understand the components that make up the cloud. Install and make an OpenStack distribution from Mirantis, Red Hat or another community version. Work with OpenStack Identity Management, Dashboard, CLI, Object Storage, Block Storage, Networking, Telemetry, Orchestration, and Image Services. Learn how to troubleshoot all the main OpenStack services. Understand where to find information for

future work with OpenStack. Who This Book Is For Certified OpenStack Administrator Study Guide is for Cloud and Linux engineers looking for a better understanding of how to work with the modern OpenStack IaaS Cloud, and wants to prove their knowledge by passing a Certified OpenStack Administrator Exam.

Software Architecture with Python IBM Redbooks

Discover the essential concepts of libvirt development and see how to interface to Linux virtualization environments, such as QEMU/KVM, XEN, Virtuozzo, VMWare ESX, LXC, Bhyve, and more. This book will prepare you to set up and maintain a virtual machine environment. You'll start by reviewing virtualization in general and then move on to libvirt-specific concepts using Python, including virtualized operating systems and networks, connections, storage pools, and event and error handling. This work concludes with a comprehensive look at the XML schema definitions for domains, networks, devices, network filtering, storage, node devices, and more. The libvirt API covers the entire life cycle of virtual objects, from creation to destruction. It contains everything needed for the management of a virtual object during that life cycle. While libvirt has APIs that support many languages, Foundations of Libvirt Development concentrates on Python exclusively, and how to use the APIs to control virtual machines under the QEMU/KVM system. and more. What You'll Learn Interface Python to the libvirt library. Review the class layout and methods of the libvirt library. Install and manipulate virtual machines via Python/libvirt. Create XML to manipulate domains, networks, and devices. Write Python programs to perform libvirt functions without human intervention. Who This Book Is ForMaintainers of virtual machines in a UNIX/Linux environment ranging from managing code on a single virtual machine through an entire installation of virtual machines.

C++ Network Programming, Volume 2 Penguin

Master the objectives required to pass the Certified OpenStack Administrator exam. About This Book Focuses on providing a clear, concise strategy so you gain the specific skills required to pass the Certified OpenStack Administrator exam Includes exercises and performance-based tasks to ensure all exam objectives can be completed via the Horizon dashboard and command-line interface Includes a free OpenStack Virtual Appliance to practice the objectives covered throughout the book Includes a practice exam to put your OpenStack skills to the test to prove you have what it takes to conquer the live exam Updated for the 2017 exam featuring OpenStack Newton Who This Book Is For This book is for IT professionals, system administrators, DevOps engineers, and software developers with basic Linux command-line and networking knowledge. It's also a great guide for those interested in an entry-level OpenStack position but have limited real-world OpenStack experience. After passing the exam, Certified OpenStack Administrators will prove they have the required skills for the job. What You Will Learn Manage the Keystone identity service by creating and modifying domains, groups, projects, users, roles, services, endpoints, and quotas. Upload Glance images, launch new Nova instances, and create flavors, key pairs, and snapshots. Discover Neutron tenant and provider networks, security groups, routers, and floating IPs. Manage the Cinder block storage service by creating volumes and attaching them to instances. Create Swift containers and set access control lists to allow read/write access to your

objects. Explore Heat orchestration templates and create, list, and update stacks. In Detail This book provides you with a specific strategy to pass the OpenStack Foundation's first professional certification: the Certified OpenStack Administrator. In a recent survey, 78% of respondents said the OpenStack skills shortage had deterred them from adopting OpenStack. Consider this an opportunity to increase employer and customer confidence by proving you have the skills required to administrate real-world OpenStack clouds. You will begin your journey by getting well-versed with the OpenStack environment, understanding the benefits of taking the exam, and installing an included OpenStack all-in-one virtual appliance so you can work through objectives covered throughout the book. After exploring the basics of the individual services, you will be introduced to strategies to accomplish the exam objectives relevant to Keystone, Glance, Nova, Neutron, Cinder, Swift, Heat, and troubleshooting. Finally, you'll benefit from the special tips section and a practice exam to put your knowledge to the test. By the end of the journey, you will be ready to become a Certified OpenStack Administrator! Style and approach Clear, concise, and straightforward with supporting diagrams and lab environment tutorials, this book will help you confidently pass Certified OpenStack Administrator objectives on the Horizon dashboard and command-line interface.

Voice of General Aviation

Independently Published

The coauthors of the New York Times–bestselling Difficult Conversations take on the toughest topic of all:

how we see ourselves Douglas Stone and Sheila Heen have spent the past fifteen years working with corporations, nonprofits, governments, and families to determine what helps us learn and what gets in our way. In Thanks for the Feedback, they explain why receiving feedback is so crucial yet so challenging, offering a simple framework and powerful tools to help us take on life's blizzard of offhand comments, annual evaluations, and unsolicited input with curiosity and grace. They blend the latest insights from neuroscience and psychology with practical, hard-headed advice. Thanks for the Feedback is destined to become a classic in the fields of leadership, organizational behavior, and education.

Learning RabbitMQ Packt Publishing Ltd

Zing! Cupid's arrow skewers a primitive part of the brain. Obediently, we fall in love amid showers of passionate fireworks, bond for a time … and then often get fed up with each other and grow irritable or numb. Perhaps we try to remodel our mate, seek solace online, or pursue a new love interest. Ancient sages recognized this biological snare and hinted at a way to dodge it: use lovemaking to balance one another and harmony arises naturally. With an entertaining blend of personal experiences, the latest neuroscience, and forgotten insights from around the globe, Cupid's Poisoned Arrow confronts current assumptions about sex and love and offers a refreshing, practical approach to sexuality.

RabbitMQ Cookbook Apress

Leverage the power of Linux to develop captivating and powerful

embedded Linux projects About This Book Explore the best practices for all embedded product development stages Learn about the compelling features offered by the Yocto Project, such as customization, virtualization, and many more Minimize project costs by using open source tools and programs Who This Book Is For If you are a developer who wants to build embedded systems using Linux, this book is for you. It is the ideal guide for you if you want to become proficient and broaden your knowledge. A basic understanding of C programming and experience with systems programming is needed. Experienced embedded Yocto developers will find new insight into working methodologies and ARM specific development competence. What You Will Learn Use the Yocto Project in the embedded Linux development process Get familiar with and customize the bootloader for a board Discover more about real-time layer, security, virtualization, CGL, and LSB See development workflows for the U-Boot and the Linux kernel, including debugging and optimization Understand the open source licensing requirements and how to comply with them when cohabiting with proprietary programs Optimize your production systems by reducing the size of both the Linux kernel and root filesystems Understand device trees and make changes to accommodate new hardware on your device Design and write multi-threaded applications using POSIX threads Measure real-time latencies and tune the Linux kernel to minimize them In Detail Embedded Linux is a complete Linux distribution employed to operate embedded devices such as smartphones, tablets, PDAs, set-top boxes, and many more. An example of an embedded Linux distribution is Android, developed by Google. This learning path starts with the module Learning Embedded Linux Using the Yocto Project. It introduces embedded Linux software and hardware architecture and presents information about the bootloader. You will go through Linux kernel

features and source code and get an overview of the Yocto Project components available. The next module Embedded Linux Projects Using Yocto Project Cookbook takes you through the installation of a professional embedded Yocto setup, then advises you on best practices. Finally, it explains how to quickly get hands-on with the Freescale ARM ecosystem and community layer using the affordable and open source Wandboard embedded board. Moving ahead, the final module Mastering Embedded Linux Programming takes you through the product cycle and gives you an in-depth description of the components and options that are available at each stage. You will see how functions are split between processes and the usage of POSIX threads. By the end of this learning path, your capabilities will be enhanced to create robust and versatile embedded projects. This Learning Path combines some of the best that Packt has to offer in one complete, curated package. It includes content from the following Packt products: Learning Embedded Linux Using the Yocto Project by Alexandru Vaduva Embedded Linux Projects Using Yocto Project Cookbook by Alex Gonzalez Mastering Embedded Linux Programming by Chris Simmonds Style and approach This comprehensive, step-by-step, pragmatic guide enables you to build custom versions of Linux for new embedded systems with examples that are immediately applicable to your embedded developments. Practical examples provide an easy-to-follow way to learn Yocto project development using the best practices and working methodologies. Coupled with hints and best practices, this will help you understand embedded Linux better.

Understanding Other Minds Manning Publications

This book comprises 26 exciting chapters by internationally renowned scholars, addressing the central psychological process separating humans

from other animals: the ability to imagine the thoughts and feelings of others, and to reflect on the contents of our own mindsa theory of mind (ToM). The four sections of the book cover developmental, cultural, and neurobiological approaches to ToM across different populations and species. The chapters explore the earliest stages of development of ToM in infancy, and how plastic ToM learning is; why 3-year-olds typically fail false belief tasks and how ToM continues to develop beyond childhood into adulthood; the debate between simulation theory and theory theory; cross-cultural perspectives on ToM and how ToM develops differently in deaf children; how we use our ToM when we make moral judgments, and the link between emotional intelligence and ToM; the neural basis of ToM measured by evoked response potentials, functional magnetic resonance imaging, and studies of brain damage; emotional vs. cognitive empathy in neuropsychiatric conditions such as autism, schizophrenia, and psychopathy; the concept of self in autism and teaching methods targeting ToM deficits; the relationship between empathy, the pain matrix and the mirror neuron system; the role of oxytocin and fetal testosterone in mentalizing and empathy; the heritability of empathy and candidate single nucleotide polymorphisms associated with empathy; and ToM in non-human primates. These 26 chapters represent a masterly

overview of a field that has deepened since the first edition was published in 1993.

Distributed Messaging for Everyone Springer Nature
This IBM® Redbooks® publication introduces the IBM Software Defined Environment (SDE) solution, which helps to optimize the entire computing infrastructure--compute, storage, and network resources--so that it can adapt to the type of work required. In today's environment, resources are assigned manually to workloads, but that happens automatically in a SDE. In an SDE, workloads are dynamically assigned to IT resources based on application characteristics, best-available resources, and service level policies so that they deliver continuous, dynamic optimization and reconfiguration to address infrastructure issues. Underlying all of this are policy-based compliance checks and updates in a centrally managed environment. Readers get a broad introduction to the new architecture. Think integration, automation, and optimization. Those are enablers of cloud delivery and analytics. SDE can accelerate business success by matching workloads and resources so that you have a responsive, adaptive environment. With the IBM Software Defined Environment, infrastructure is fully programmable to rapidly deploy workloads on optimal resources and to instantly respond to changing business demands. This information is intended for IBM sales representatives, IBM software architects, IBM Systems Technology Group brand specialists, distributors, resellers, and anyone who is developing or implementing SDE.
Simon and Schuster
If you want to push your Java skills to the next level, this book provides expert advice from Java leaders and practitioners. You'll be encouraged to look at problems in new ways, take broader responsibility for your work, stretch yourself by learning new techniques, and

become as good at the entire craft of development as you possibly can. Edited by Kevlin Henney and Trisha Gee, 97 Things Every Java Programmer Should Know reflects lifetimes of experience writing Java software and living with the process of software development. Great programmers share their collected wisdom to help you rethink Java practices, whether working with legacy code or incorporating changes since Java 8. A few of the 97 things you should know: "Behavior Is Easy, State Is Hard"—Edson Yanaga “Learn Java Idioms and Cache in Your Brain”—Jeanne Boyarsky “Java Programming from a JVM Performance Perspective”—Monica Beckwith "Garbage Collection Is Your Friend"—Holly K Cummins “Java's Unspeakable Types”—Ben Evans "The Rebirth of Java"—Sander Mak “Do You Know What Time It Is?”—Christin Gorman

RabbitMQ in Depth North Atlantic Books

Build, customize, and debug your own Android system About This Book Master Android system-level programming by integrating, customizing, and extending popular open source projects Use Android emulators to explore the true potential of your hardware Master key debugging techniques to create a hassle-free development environment Who This Book Is For This book is for Android system programmers and developers who want to use Android and create indigenous projects with it. You should know the important points about the operating system and the C/C++ programming language. What You Will Learn Set up the Android

development environment and organize source code repositories Get acquainted with the Android system architecture Build the Android emulator from the AOSP source tree Find out how to enable WiFi in the Android emulator Debug the boot up process using a customized Ramdisk Port your Android system to a new platform using VirtualBox Find out what recovery is and see how to enable it in the AOSP build Prepare and test OTA packages In Detail Android system programming involves both hardware and software knowledge to work on system level programming. The developers need to use various techniques to debug the different components in the target devices. With all the challenges, you usually have a deep learning curve to master relevant knowledge in this area. This book will not only give you the key knowledge you need to understand Android system programming, but will also prepare you as you get hands-on with projects and gain debugging skills that you can use in your future projects. You will start by exploring the basic setup of AOSP, and building and testing an emulator image. In the first project, you will learn how to customize and extend the Android emulator. Then you'll move on to the real challenge—building your own Android system on VirtualBox. You'll see how to debug the init process, resolve the bootloader issue, and enable various hardware interfaces. When you have a complete system, you will learn how to patch and

upgrade it through recovery. Throughout the book, you will get to know useful tips on how to integrate and reuse existing open source projects such as LineageOS (CyanogenMod), Android-x86, Xposed, and GApps in your own system. Style and approach This is an easy-to-follow guide full of hands-on examples and system-level programming tips.

IBM Software Defined Environment Packt Publishing Ltd

Build and optimize efficient messaging applications with ease About This Book Learn to administer, configure, and manage RabbitMQ instances Discover ways to secure and troubleshoot RabbitMQ instances This book is fully up-to-date with all the latest changes to version 3.5 Who This Book Is For If you are a developer or system administrator with a basic knowledge of messaging who wants to learn RabbitMQ, or if you want to further enhance your knowledge in working with the message broker, then this book is ideal for you. To fully understand some examples in the book, a basic knowledge of the Java programming language is required. What You Will Learn Apply messaging patterns using the message broker Administer RabbitMQ using the command line, management Web console, or management REST services Create a cluster of scalable, and highly-available, RabbitMQ instances Use RabbitMQ with the Spring Framework, MuleESB, WSO2, and Oracle databases Deploy RabbitMQ using Puppet, Vagrant, or Docker Fine-tune the performance of RabbitMQ Monitor RabbitMQ using Nagios, Munin, or Monit Secure, troubleshoot, and extend RabbitMQ In Detail

RabbitMQ is Open Source Message Queuing software based on the Advanced Message Queue Protocol Standard written in the Erlang Language. RabbitMQ is an ideal candidate for large-scale projects ranging from e-commerce and finance to Big Data and social networking because of its ease of use and high performance. Managing RabbitMQ in such a dynamic environment can be a challenging task that requires a good understanding not only of how to work properly with the message broker but also of its best practices and pitfalls. Learning RabbitMQ starts with a concise description of messaging solutions and patterns, then moves on to concrete practical scenarios for publishing and subscribing to the broker along with basic administration. This knowledge is further expanded by exploring how to establish clustering and high availability at the level of the message broker and how to integrate RabbitMQ with a number of technologies such as Spring, and enterprise service bus solutions such as MuleESB and WSO2. We will look at advanced topics such as performance tuning, secure messaging, and the internals of RabbitMQ. Finally we will work through case-studies so that we can see RabbitMQ in action and, if something goes wrong, we'll learn to resolve it in the Troubleshooting section. Style and approach Each chapter of the book is an easy-to-follow guide that expands and builds on the knowledge already gained in previous chapters. Throughout the course of the book, a sample system called the CSN (Corporate Social Network) is used to illustrate the core principles described. At the end of each chapter, there is a Q&A session that covers practical questions that may

arise in practice when working with RabbitMQ.

The ACE Programmer's Guide Addison-Wesley Professional

Tap into the wisdom of experts to learn what every programmer should know, no matter what language you use. With the 97 short and extremely useful tips for programmers in this book, you'll expand your skills by adopting new approaches to old problems, learning appropriate best practices, and honing your craft through sound advice. With contributions from some of the most experienced and respected practitioners in the industry--including Michael Feathers, Pete Goodliffe, Diomidis Spinellis, Cay Horstmann, Verity Stob, and many more--this book contains practical knowledge and principles that you can apply to all kinds of projects. A few of the 97 things you should know: "Code in the Language of the Domain" by Dan North "Write Tests for People" by Gerard Meszaros "Convenience Is Not an -ility" by Gregor Hohpe "Know Your IDE" by Heinz Kabutz "A Message to the Future" by Linda Rising "The Boy Scout Rule" by Robert C. Martin (Uncle Bob) "Beware the Share" by Udi Dahan

Flexible Software Architecture O'Reilly Media

This beautiful little gift book featuring The Very Hungry Caterpillar provides a simple, easy-to-follow journey to relaxation, perfect for children and adults. Just stop and breathe . . . How do you feel? Join The Very Hungry Caterpillar for a soothing story featuring classic art from the World of Eric Carle. With simple coping mechanisms and beautiful illustrations, this book is an excellent reminder to kids and adults alike to take a moment, breathe deep, and smile

Developing Interoperable and Federated Cloud Architecture "O'Reilly Media, Inc."

The field of agent and multi-agent systems is concerned with the development and evaluation of sophisticated,

AI-based, problem solving and control architectures for both single and multi-agent systems. This book presents the proceedings of the 7th KES Conference on Agent and Multi-agent Systems – Technologies and Applications (KES-AMSTA 2013), held in Hue City, Vietnam, in May 2013. The KES-AMSTA 2013 conference provides an internationally respected forum for scientific research in the technologies and applications of agent and multi-agent systems. In all, 44 papers were selected for oral presentation and publication in this volume. Special attention is paid to the feature topics of intelligent technologies and applications in the area of e-health, social networking, self-organizing systems, economics and trust management. Other topics covered include: agent oriented software engineering; beliefs engineering; desires and intentions representation; agent cooperation, coordination, negotiation, organization and communication; distributed problem-solving; specification of agent communication languages; formalization of ontologies; and conversational agents. The book highlights new trends and challenges in agent and multi-agent research, and will be of interest to the research community working in the fields of artificial intelligence, collective computational intelligence, robotics, dialogue systems and, in particular, agent and multi-agent systems, technologies and

applications.

Design modern systems using effective architecture concepts, design patterns, and techniques with C++20

Addison-Wesley Professional

Apply business requirements to IT infrastructure and deliver a high-quality product by understanding architectures such as microservices, DevOps, and cloud-native using modern C++ standards and features Key Features Design scalable large-scale applications with the C++ programming language Architect software solutions in a cloud-based environment with continuous integration and continuous delivery (CI/CD) Achieve architectural goals by leveraging design patterns, language features, and useful tools Book Description Software architecture refers to the high-level design of complex applications. It is evolving just like the languages we use. Modern C++ allows developers to write high-performance apps in a high-level language without sacrificing readability and maintainability. If you're working with modern C++, this practical guide will help you put your knowledge to work and design distributed, large-scale apps. You'll start by getting up to speed with architectural concepts, including established patterns and rising trends. The book will then explain what software architecture is and help you explore its components. Next, you'll discover the design concepts involved in application architecture and the patterns in software development, before going on to learn how to build, package, integrate, and deploy your components. In the concluding chapters, you'll explore different architectural qualities, such as maintainability, reusability, testability, performance, scalability, and

security. Finally, you will get an overview of distributed systems, such as service-oriented architecture, microservices, and cloud-native, and understand how to apply them in application development. By the end of this book, you'll be able to build distributed services using modern C++ and associated tools to deliver solutions as per your clients' requirements. What you will learn Understand how to apply the principles of software architecture Apply design patterns and best practices to meet your architectural goals Write elegant, safe, and performant code using the latest C++ features Build applications that are easy to maintain and deploy Explore the different architectural approaches and learn to apply them as per your requirement Simplify development and operations using application containers Discover various techniques to solve common problems in software design and development Who this book is for This software architecture C++ programming book is for experienced C++ developers who are looking to become software architects or are interested in developing enterprise-grade applications.

Perspectives from developmental social neuroscience Advanced Methods and Technologies for Agent and Multi-Agent Systems

Microservices is an architectural style in which large, complex software applications are composed of one or more smaller services. Each of these microservices focuses on completing one task that represents a small business capability. These microservices can be developed in any programming language.

This IBM® Redbooks® publication covers Microservices best practices for Java. It focuses on creating cloud native applications using the latest version of IBM WebSphere® Application Server Liberty, IBM Bluemix® and other Open Source Frameworks in the Microservices ecosystem to highlight Microservices best practices for Java.

The Science and Art of Receiving Feedback Well

Packt Publishing Ltd

Summary RabbitMQ in Action is a fast-paced run through building and managing scalable applications using the RabbitMQ messaging server. It starts by explaining how message queuing works, its history, and how RabbitMQ fits in. Then it shows you real-world examples you can apply to your own scalability and interoperability challenges. About the Technology There's a virtual switchboard at the core of most large applications where messages race between servers, programs, and services. RabbitMQ is an efficient and easy-to-deploy queue that handles this message traffic effortlessly in all situations, from web startups to massive enterprise systems. About the Book RabbitMQ in Action teaches you to build and manage scalable applications in multiple languages using the RabbitMQ messaging server. It's a snap to get started. You'll learn how message queuing works and how RabbitMQ fits in. Then, you'll explore practical scalability and interoperability issues through many examples. By the end, you'll know how to make Rabbit run like a well-oiled machine in a 24 x 7 x 365 environment. Written for developers familiar with Python, PHP, Java, .NET, or

any other modern programming language. No RabbitMQ experience required. Purchase of the print book comes with an offer of a free PDF, ePub, and Kindle eBook from Manning. Also available is all code from the book. What's Inside Learn fundamental messaging design patterns Use patterns for on-demand scalability Glue a PHP frontend to a backend written in anything Implement a PubSub-alerting service in 30 minutes flat Configure RabbitMQ's built-in clustering Monitor, manage, extend, and tune RabbitMQ ========== ==========?==== Table of Contents Pulling RabbitMQ out of the hat Understanding messaging Running and administering Rabbit Solving problems with Rabbit: coding and patterns Clustering and dealing with failure Writing code that survives failure Warrens and Shovels: failover and replication Administering RabbitMQ from the Web Controlling Rabbit with the REST API Monitoring: Houston, we have a problem Supercharging and securing your Rabbit Smart Rabbits: extending RabbitMQ

How to Set Up and Maintain a Virtual Machine Environment with Python Springer

The software profession has a problem, widely recognized but which nobody seems willing to do anything about; a variant of the well known ""telephone game,"" where some trivial rumor is repeated from one person to the next until it has become distorted beyond recognition and blown up out of all proportion. Unfortunately, the objects of this telephone game are generally considered cornerstone

truths of the discipline, to the point that their acceptance now seems to hinder further progress. This book takes a look at some of those ""ground truths"" the claimed 10x variation in productivity between developers; the ""software crisis""; the cost-of-change curve; the ""cone of uncertainty""; and more. It assesses the real weight of the evidence behind these ideas - and confronts the scary prospect of moving the state of the art forward in a discipline that has had the ground kicked from under it.