



Developer Handbook

A Developer Handbook covering a variuous of Computer Architectures, Programming languages, Security, Networking, Game Developement, Graphics programming, Agile Development, etc.. This handbook also includes the Applications, Frameworks, Libraries and Tools that will make you a better and more efficient developer within each of Development.

HandBook Version 1.0, 2021

[View License](#)

[Contributing](#)

Note 1: This Developer handbook will be continuously updated and improved with new topics, diagrams, code samples, and other content throughout 2021 & beyond. This handook will also be available in other document formats such as PDF, [AsciiDoc](#) using [Asciidoctor](#) and [MkDocs](#), which is a fast and simple static site generator that's geared towards building project documentation.

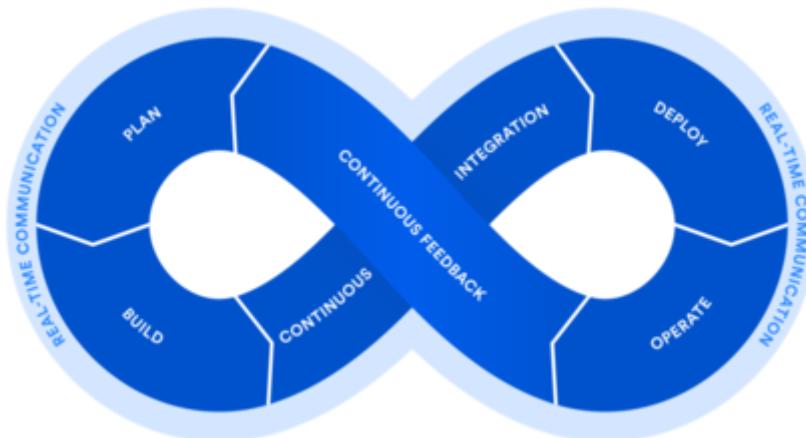
Note 2: You can easily convert this markdown file to a PDF in [VSCode](#) using this handy extension [Markdown PDF](#).

Table of Contents

Section 1	Section 2	Section 3
1. DevOps	21. Swift	41. Objective-C
2. IoT	22. Golang	42. Qt
3. Open Source Security	23. C++	43. LabVIEW
4. Linux	24. Ruby	44. Bootstrap
5. Networking	25. Java	45. Elixir
6. Cloud-Native	26. Scala	46. Erlang
7. Kubernetes	27. Groovy	47. Lua
8. ARM	28. Clojure	48. Vala
9. Assembly(x86)	29. WebAssembly	49. Haskell
10. Machine Learning	30. CUDA	50. PHP
11. Python Guide	31. Julia	51. Robotics
12. SQL	32. R	52. WSL
13. HMTL/CSS	33. MATLAB	53. 3D Graphics & Design
14. React	34. Bash/Shell/PowerShell	54. Game Development
15. Angular	35. C#	55. Blockchain
16. VueJS	36. F#	56. Working Remote
17. Node.js	37. .NET/.NET Core	57. Audio & Video Editing
18. TypeScript	38. Dart	58. Podcasting
19. Rust	39. Flutter	59. Agile Development
20. Kotlin	40. Firebase	60. RISC-V
61. Serverless	62. XML	63. Verilog/SystemVerilog
64. Quantum Computing	65. Data Structures & Algorithms	66. Differential Privacy
67. Raspberry Pi	68. Arduino	69. FPGA
70. Wayland	71. Electron	72. jQuery
73. FinOps	74. Telco 5G	

1. DevOps

[Back to the Top](#)



DevOps Guide

A guide covering DevOps including the applications and tools that will make you a better and more efficient DevOps Engineer.

Resources

[DevOps Engineering on AWS from AWS Training](#)

[AWS Certified DevOps Engineer - Professional from A Cloud Guru](#)

[Microsoft Certified: DevOps Engineer Expert Cert.](#)

[Introduction to Azure DevOps from A Cloud Guru](#)

[Architecting with Google Compute Engine](#)

[Architecting with Google Kubernetes Engine in Google Cloud](#)

[VMware Training and Certification Program](#)

[Cloudera Certification Program](#)

[Salesforce Certification Program](#)

[Salesforce Superbadges](#)

[Red Hat Training and Certification Program](#)

[Linux Foundation Training and Certification Program](#)

[Linux Professional Institute\(LPI\) Training and Certification](#)

[Learn DevOps with Online Courses and Lessons from edX](#)

[Top DevOps Courses Online from Udemy](#)

Devops Courses from Coursera

[Snowflake Hands-On Essentials Online Course](#)

[Snowflake Education and Training](#)

[Snowflake Certifications](#)

[Data Cloud Academy from Snowflake](#)

[Snowflake Decoded: Fundamentals and hands on Training on Udemy](#)

Tools

[GitHub](#) provides hosting for software development version control using Git. It offers all of the distributed version control and source code management functionality of Git as well as adding its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project.

[GitHub Codespaces](#) is an integrated development environment(IDE) on GitHub. That allows developers to develop entirely in the cloud using Visual Studio and Visual Studio Code.

[GitHub Actions](#) will automate, customize, and execute your software development workflows right in your repository with GitHub Actions. You can discover, create, and share actions to perform any job you'd like, including CI/CD, and combine actions in a completely customized workflow.[GitHub Actions for Azure](#) you can create workflows that you can set up in your repository to build, test, package, release and deploy to Azure.Learn more about all other integrations with Azure.

[GitLab](#) is a web-based DevOps lifecycle tool that provides a Git-repository manager providing wiki, issue-tracking and CI/CD pipeline features, using an open-source license, developed by GitLab Inc.

[Jenkins](#) is a free and open source automation server. Jenkins helps to automate the non-human part of the software development process, with continuous integration and facilitating technical aspects of continuous delivery.

[Bitbucket](#) is a web-based version control repository hosting service owned by Atlassian, for source code and development projects that use either Mercurial or Git revision control systems. Bitbucket offers both commercial plans and free accounts. It offers free accounts with an unlimited number of private repositories. Bitbucket integrates with other Atlassian software like Jira, HipChat, Confluence and Bamboo.

[Bamboo](#) is a continuous integration (CI) server that can be used to automate the release management for a software application, creating a continuous delivery pipeline.

[Codecov](#) is the leading, dedicated code coverage solution. It provides highly integrated tools to group, merge, archive and compare coverage reports. Whether your team is comparing changes in a pull request or reviewing a single commit, Codecov will improve the code review workflow and quality.

[Drone](#) is a Continuous Delivery system built on container technology. Drone uses a simple YAML configuration file, a superset of docker-compose, to define and execute Pipelines inside Docker containers.

[Travis CI](#) is a hosted continuous integration service used to build and test software projects hosted at GitHub.

[Circle CI](#) is a continuous integration and continuous delivery platform that helps software teams work smarter, faster.

[Zuul-CI](#) is a program that drives continuous integration, delivery, and deployment systems with a focus on project gating and interrelated projects. Using the same [Ansible playbooks](#) to deploy your system and run your tests.

[Artifactory](#) is a Universal Artifact Repository Manager developed by JFrog. It supports all major packages, enterprise ready security, clustered, HA, Docker registry, multi-site replication and scalable.

[Azure DevOps](#) is a set of services for teams to share code, track work, and ship software; CLIs Build, deploy, diagnose, and manage multi-platform, scalable apps and services; Azure Pipelines Continuously build, test, and deploy to any platform and cloud; Azure Lab Services Set up labs for classrooms, trials, development and testing, and other scenarios.

[Team City](#) is a build management and continuous integration server from JetBrains.

[Shippable](#) simplifies DevOps and makes it systematic with an Assembly Line platform that is heterogeneous, flexible, and provides complete visibility across your DevOps workflows.

[Spinnaker](#) is an open source, multi-cloud continuous delivery platform for releasing software changes with high velocity and confidence.

[AWS CodeBuild](#) is a fully managed continuous integration service that compiles source code, runs tests, and produces software packages that are ready to deploy. With CodeBuild, you don't need to provision, manage, and scale your own build servers.

[Selenium](#) is a free (open source) automated testing suite for web applications across different browsers and platforms.

[Cucumber](#) is a tool based on Behavior Driven Development (BDD) framework which is used to write acceptance tests for the web application. It allows automation of functional validation in easily readable and understandable format (like plain English) to Business Analysts, Developers, and Testers.

[JUnit](#) is a unit testing framework for the Java programming language.

[Mocha](#) is a JavaScript test framework for Node.js programs, featuring browser support, asynchronous testing, test coverage reports, and use of any assertion library.

[Karma](#) is a simple tool that allows you to execute JavaScript code in multiple real browsers.

[Jasmine](#) is an open source testing framework for JavaScript. It aims to run on any JavaScript-enabled platform, to not intrude on the application nor the IDE, and to have easy-to-read syntax.

[Maven](#) is a build automation tool used primarily for Java projects. Maven can also be used to build and manage projects written in C#, Ruby, Scala, and other languages. The Maven project is hosted by the Apache Software Foundation.

[Gradle](#) is an open-source build-automation system that builds upon the concepts of Apache Ant and Apache Maven and introduces a Groovy-based domain-specific language instead of the XML form used by Apache Maven for declaring the project configuration.

[Chef](#) is an effortless Infrastructure Suite offers visibility into security and compliance status across all infrastructure and makes it easy to detect and correct issues long before they reach production.

[Puppet](#) is an open source tool that makes continuous integration and delivery of your software on traditional or containerized infrastructure easy by pulling together all your existing tools and giving you flexibility to deploy your way.

[Ansible](#) is an open-source software provisioning, configuration management, and application-deployment tool. It runs on many Unix-like systems, and can configure both Unix-like systems as well as Microsoft Windows.

[Kubelnit](#) provides Ansible playbooks and roles for the deployment and configuration of multiple Kubernetes distributions.

[Salt](#) is Python-based, open-source software for event-driven IT automation, remote task execution, and configuration management. Supporting the "Infrastructure as Code" approach to data center system and network deployment and management, configuration automation, SecOps orchestration, vulnerability remediation, and hybrid cloud control.

[Terraform](#) is an open-source infrastructure as code software tool created by HashiCorp. It enables users to define and provision a datacenter infrastructure using a high-level configuration language known as Hashicorp Configuration Language (HCL), or optionally JSON.

[Consul](#) is a service networking solution to connect and secure services across any runtime platform and public or private cloud.

[Packer](#) is lightweight, runs on every major operating system, and is highly performant, creating machine images for multiple platforms in parallel. Packer does not replace configuration management like Chef or Puppet. In fact, when building images, Packer is able to use tools like Chef or Puppet to install software onto the image.

[Nomad](#) is a highly available, distributed, data-center aware cluster and application scheduler designed to support the modern datacenter with support for long-running services, batch jobs, and much more.

[Vagrant](#) is a tool for building and managing virtual machine environments in a single workflow. With an easy-to-use workflow and focus on automation, Vagrant lowers development environment setup time and increases production parity.

[Vault](#) is a tool for securely accessing secrets. A secret is anything that you want to tightly control access to, such as API keys, passwords, certificates, and more. Vault provides a unified interface to any secret, while providing tight access control and recording a detailed audit log.

[CFEngine](#) is an open-source configuration management system, written by Mark Burgess. Its primary function is to provide automated configuration and maintenance of large-scale computer systems, including the unified management of servers, desktops, consumer and industrial devices, embedded networked devices, mobile smartphones, and tablet computers.

[Octpus Deploy](#) is the deployment automation server for your entire team, designed to make it easy to orchestrate releases and deploy applications, whether on-premises or in the cloud.

[AWS CodeDeploy](#) is a fully managed deployment service that automates software deployments to a variety of compute services such as Amazon EC2, AWS Fargate, AWS Lambda, and your on-premises servers. AWS CodeDeploy makes it easier for you to rapidly release new features, helps you avoid downtime during application deployment, and handles the complexity of updating your applications.

[Kubernetes](#) is an open-source container-orchestration system for automating application deployment, scaling, and management. It was originally designed by Google, and is now maintained by the Cloud Native Computing Foundation.

[Docker](#) is a set of platform as a service products that use OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries and configuration files; they can communicate with each other through well-defined channels. All containers are run by a single operating-system kernel and are thus more lightweight than virtual machines.

[PowerShell/PowerShell Core](#) is a cross-platform (Windows, Linux, and macOS) automation and configuration tool/framework that works well with your existing tools and is optimized for dealing with structured data (e.g. JSON, CSV, XML, etc.), REST APIs, and object models. It includes a command-line shell, an associated scripting language and a framework for processing cmdlets.

[Hyper-V](#) creates virtual machines on Windows 10. Hyper-V can be enabled in many ways including using the Windows 10 control panel, PowerShell or using the Deployment Imaging Servicing and Management tool (DISM).

[Cloud Hypervisor](#) is an open source Virtual Machine Monitor (VMM) that runs on top of [KVM](#). The project focuses on exclusively running modern, cloud workloads, on top of a limited set of hardware architectures and platforms. Cloud workloads refers to those that are usually run by customers inside a cloud provider. Cloud Hypervisor is implemented in [Rust](#) and is based on the [rust-vmm](#) crates.

[Intel® Graphics Virtualization Technology \(Intel® GVT\)](#) is a full GPU virtualization solution with mediated pass-through, starting from 4th generation Intel Core (TM) processors with Intel processor graphics(Broadwell and newer). It can be used to virtualize the GPU for multiple guest virtual machines, effectively providing near-native graphics performance in the virtual machine and still letting your host use the virtualized GPU normally.

[VMware vSphere Hypervisor](#) is a bare-metal hypervisor that virtualizes servers; allowing you to consolidate your applications while saving time and money managing your IT infrastructure.

[VMware vSphere](#) is the industry-leading compute virtualization platform, and your first step to application modernization. It has been rearchitected with native Kubernetes to allow customers to modernize the 70 million+ workloads now running on vSphere.

[VMware Tanzu](#) is a centralized management platform for consistently operating and securing your Kubernetes infrastructure and modern applications across multiple teams and private/public clouds.

[Rancher](#) is a complete software stack for teams adopting containers. It addresses the operational and security challenges of managing multiple Kubernetes clusters, while providing DevOps teams with integrated tools for running containerized workloads.

[K3s](#) is a highly available, certified Kubernetes distribution designed for production workloads in unattended, resource-constrained, remote locations or inside IoT appliances.

[Rook](#) is an open source cloud-native storage orchestrator for Kubernetes that turns distributed storage systems into self-managing, self-scaling, self-healing storage services. It automates the tasks of a storage administrator: deployment, bootstrapping, configuration, provisioning, scaling, upgrading, migration, disaster recovery, monitoring, and resource management.

[Google Kubernetes Engine \(GKE\)](#) is a managed, production-ready environment for deploying containerized applications.

[Anthos](#) is a modern application management platform that provides a consistent development and operations experience for cloud and on-premises environments.

[AWS ECS](#) is a highly scalable, high-performance container orchestration service that supports Docker containers and allows you to easily run and scale containerized applications on AWS. Amazon ECS eliminates the need for you to install and operate your own container orchestration software, manage and scale a cluster of virtual machines, or schedule containers on those virtual machines.

[Apache Mesos](#) is a cluster manager that provides efficient resource isolation and sharing across distributed applications, or frameworks. It can run Hadoop, Jenkins, Spark, Aurora, and other frameworks on a dynamically shared pool of nodes.

[Apache Spark](#) is a unified analytics engine for big data processing, with built-in modules for streaming, SQL, machine learning and graph processing.

[Apache Hadoop](#) is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

[Microsoft Azure](#) is a cloud computing service created by Microsoft for building, testing, deploying, and managing applications and services through Microsoft-managed data centers.

[Azure Functions](#) is a solution for easily running small pieces of code, or "functions," in the cloud. You can write just the code you need for the problem at hand, without worrying about a whole application or the infrastructure to run it.

[Rkt](#) is a pod-native container engine for Linux. It is composable, secure, and built on standards.

[AWS Lambda](#) is an event-driven, serverless computing platform provided by Amazon as a part of the Amazon Web Services. It is a computing service that runs code in response to events and automatically manages the computing resources required by that code.

[Helm](#) is the Kubernetes Package Manager.

[Kubespray](#) is a tool that combines Kubernetes and Ansible to easily install Kubernetes clusters that can be deployed on [AWS](#), [GCE](#), [Azure](#), [OpenStack](#), [vSphere](#), [Packet](#) (bare metal), Oracle Cloud Infrastructure (Experimental), or Baremetal

[Red Hat OpenShift](#) is focused on security at every level of the container stack and throughout the application lifecycle. It includes long-term, enterprise support from one of the leading Kubernetes contributors and open source software companies.

[OpenShift Hive](#) is an operator which runs as a service on top of Kubernetes/OpenShift. The Hive service can be used to provision and perform initial configuration of OpenShift 4 clusters.

[OKD](#) is a community distribution of Kubernetes optimized for continuous application development and multi-tenant deployment. OKD adds developer and operations-centric tools on top of Kubernetes to enable rapid application development, easy deployment and scaling, and long-term lifecycle maintenance for small and large teams.

[Odo](#) is a fast, iterative, and straightforward CLI tool for developers who write, build, and deploy applications on Kubernetes and OpenShift.

[Kata Operator](#) is an operator to perform lifecycle management (install/upgrade/uninstall) of [Kata Runtime](#) on Openshift as well as Kubernetes cluster.

[Knative](#) is a Kubernetes-based platform to build, deploy, and manage modern serverless workloads. Knative takes care of the operational overhead details of networking, autoscaling (even to zero), and revision tracking.

[Etcd](#) is a distributed key-value store that provides a reliable way to store data that needs to be accessed by a distributed system or cluster of machines. Etcd is used as the backend for service discovery and stores cluster state and configuration for Kubernetes.

[OpenStack](#) is a free and open-source software platform for cloud computing, mostly deployed as infrastructure-as-a-service that controls large pools of compute, storage, and networking resources throughout a datacenter, managed through a dashboard or via the OpenStack API. OpenStack works with popular enterprise and open source technologies making it ideal for heterogeneous infrastructure.

[Cloud Foundry](#) is an open source, multi cloud application platform as a service that makes it faster and easier to build, test, deploy and scale applications, providing a choice of clouds, developer frameworks, and application services. It is an open source project and is available through a variety of private cloud distributions and public cloud instances.

[Splunk](#) software is used for searching, monitoring, and analyzing machine-generated big data, via a Web-style interface.

[Prometheus](#) is a free software application used for event monitoring and alerting. It records real-time metrics in a time series database (allowing for high dimensionality) built using a HTTP pull model, with flexible queries and real-time alerting.

[Loki](#) is a horizontally-scalable, highly-available, multi-tenant log aggregation system inspired by Prometheus. It is designed to be very cost effective and easy to operate. It does not index the contents of the logs, but rather a set of labels for each log stream.

[Thanos](#) is a set of components that can be composed into a highly available metric system with unlimited storage capacity, which can be added seamlessly on top of existing Prometheus deployments.

[Container Storage Interface \(CSI\)](#) is an API that lets container orchestration platforms like Kubernetes seamlessly communicate with stored data via a plug-in.

[OpenEBS](#) is a Kubernetes-based tool to create stateful applications using Container Attached Storage.

[ElasticSearch](#) is a search engine based on the Lucene library. It provides a distributed, multitenant-capable full-text search engine with an HTTP web interface and schema-free JSON documents. Elasticsearch is developed in Java.

[Logstash](#) is a tool for managing events and logs. When used generically, the term encompasses a larger system of log collection, processing, storage and searching activities.

[Kibana](#) is an open source data visualization plugin for Elasticsearch. It provides visualization capabilities on top of the content indexed on an Elasticsearch cluster. Users can create bar, line and scatter plots, or pie charts and maps on top of large volumes of data.

[New Relic](#) is a SaaS-based monitoring tool that fully supports the way DevOps teams work in the modern enterprise by streamlining your workflows with today's collaboration software and orchestration tools like Puppet, Chef, and Ansible.

[Nagios](#) is a free and open source computer-software application that monitors systems, networks and infrastructure. Nagios offers monitoring and alerting services for servers, switches, applications and services. It alerts users when things go wrong and alerts them a second time when the problem has been resolved.

[SonarQube](#) is an open-source platform developed by SonarSource for continuous inspection of code quality to perform automatic reviews with static analysis of code to detect bugs, code smells, and security vulnerabilities on 20+ programming languages.

[Genie](#) is a federated job orchestration engine developed by Netflix. Genie provides REST APIs to run a variety of big data jobs like Hadoop, Pig, Hive, Spark, Presto, Sqoop and more. It also provides APIs for managing the metadata of many distributed processing clusters and the commands and applications which run on them.

[Inviso](#) is a lightweight tool that provides the ability to search for Hadoop jobs, visualize the performance, and view cluster utilization.

[Fenzo](#) is a scheduler Java library for Apache Mesos frameworks that supports plugins for scheduling optimizations and facilitates cluster autoscaling.

[Dynomite](#) is a thin, distributed dynamo layer for different storage engines and protocols, which includes [Redis](#) and [Memcached](#). Dynomite supports multi-datacenter replication and is designed for High Availability(HA).

[Dyno](#) is a tool that is used to scale a Java client application utilizing [Dynomite](#).

[Raigad](#) is a process/tool that runs alongside Elasticsearch to automate backup/recovery, Deployments and Centralized Configuration management.

[Priam](#) is a process/tool that runs alongside Apache Cassandra to automate backup/recovery, Deployments and Centralized Configuration management.

[Chaos Monkey](#) is a resiliency tool used to randomly terminates virtual machine instances and containers that run inside of your production environment. Chaos Monkey should work with any backend that [Spinnaker](#) supports (AWS, Google Compute Engine, Microsoft Azure, Kubernetes, and Cloud Foundry).

[Falcor](#) is a JavaScript library for efficient data fetching. Falcor lets you represent all your remote data sources as a single domain model via a virtual JSON graph, whether in memory on the client or over the network on the server.

[Restify](#) is a framework, utilizing [connect](#) style middleware for building REST APIs.

[Traefik](#) is an open source Edge Router that makes publishing your services a fun and easy experience. It receives requests on behalf of your system and finds out which components are responsible for handling them. What sets Traefik apart, besides its many features, is that it automatically discovers the right configuration for your services.

[Jira](#) is a proprietary issue tracking product developed by Atlassian that allows bug tracking and agile project management.

[Pivotal Tracker](#) is the agile project management tool of choice for developers around the world for real-time collaboration around a shared, prioritized backlog.

[Trello](#) is a web-based Kanban-style list-making application that gives you perspective over all your projects, at work and at home.

[Microsoft Teams](#) is the hub for team collaboration in Office 365 that integrates the people, content, and tools your team needs to be more engaged and effective.

[Slack](#) is a cloud-based proprietary instant messaging platform developed by Slack Technologies.

[OpsGenie](#) is a cloud-based service for dev & ops teams, providing reliable alerts, on-call schedule management and escalations. OpsGenie integrates with monitoring tools & services, ensures the right people are notified.

[Pagerduty](#) automates processes built on best practices, allowing you to focus on higher value parts of incident response. Granular and scalable permissions enable teams to administer and operate independently while controlling visibility.

[Veracode](#) is a leading provider of enterprise-class application security, seamlessly integrating agile security solutions for organizations around the globe. In addition to application security services and secure devops services, Veracode provides a full security assessment to ensure your website and applications are secure, and ensures full enterprise data protection.

2. IoT

[Back to the Top](#)



IoT Guide

A guide for getting started with IoT devices including software and hardware that will make you a better and more efficient IoT developer.

Resources

[Microsoft Certified: Azure IoT Developer Specialty](#)

[AWS Internet of Things Foundation Series Training](#)

[Google's Internet of Things \(IoT\)](#)

[Internet Of Things Courses from Udemy](#)

[IoT \(Internet of Things\) Courses from edX](#)

[IoT \(Internet of Things\) Courses from Coursera](#)

Tools

[Azure IoT Tools](#) is a collection of Microsoft-managed cloud services that connect, monitor, and control billions of IoT assets. In simpler terms, an IoT solution is made up of one or more IoT devices that communicate with one or more back-end services hosted in the cloud.

[AWS IoT Tools](#) has broad and deep IoT services, from the edge to the cloud. Along with IoT analytics that enables you to apply machine learning to your IoT data with hosted Jupyter Notebooks. You can directly connect your IoT data to the notebook and build, train, and execute models right from the AWS IoT Analytics console without having to manage any of the underlying infrastructure.

[Google Cloud IoT](#) is a complete set of tools to connect, process, store, and analyze data both at the edge and in the cloud.

[DeviceHive](#) is a free, highly scalable open-source IoT platform for data collection, processing and analysis, visualization, and device management with the broad range of integration options.

[Distributed Services Architecture \(DSA\)](#) is an open source IoT platform that facilitates device inter-communication, logic and applications at every layer of the Internet of Things infrastructure. The objective is to unify the disparate devices, services and applications into a structured and adaptable real-time data model.

[IoTivity](#) is an open source software framework enabling seamless device-to-device connectivity to address the emerging needs of the Internet of Things.

[InfluxDB](#) is an open source time series database, purpose-built by InfluxData for monitoring metrics and events, provides real-time visibility into stacks, sensors, and systems. Use InfluxDB to capture, analyze, and store millions of points per second, meet demanding SLA's, and chart a path to automation.

[Eclipse IoT Project](#) provides open source technology that will be used to build IoT solutions for industry and consumers.

[M2MLabs MainSpring](#) is an application framework for building machine-to-machine applications like vehicle tracking or machine remote monitoring. In such applications typically a remote device equipped with

sensors (e.g. gps, temperature, pressure) and actors communicates with a server application that is running the device communication protocol, device configuration, storage of data sent by the devices as well as the application business logic and the presentation layer.

[EdgeX Foundry](#) is a vendor-neutral project under the Linux Foundation. The initiative is aligned around a common goal: the simplification and standardization of the foundation for edge computing architectures in the Industrial IoT market, while still allowing the ecosystem to add significant value.

[The Open Connectivity Foundation](#) is dedicated to ensuring secure interoperability for consumers, businesses and industries by delivering a standard communications platform, a bridging specification, an open source implementation and a certification program allowing devices to communicate regardless of form factor, operating system, service provider, transport technology or ecosystem.

[Eclipse Foundation](#) provides our global community of individuals and organizations with a mature, scalable and commercially-friendly environment for open source software collaboration and innovation.

[Open Source Hardware Association \(OSHWA\)](#) is a non-profit organization that advocates for open-source hardware. It aims to act as a hub of open source hardware activity of all types while actively cooperating with other initiatives such as the TAPR Open Hardware License, open-source development groups at CERN, and the Open Source Initiative (OSI).

Protocols

[DBus](#) is an open source software bus developed Red Hat for inter-process communication, and remote procedure call mechanism that allows communication between multiple processes running concurrently on the same machine.

[SOAP](#) is a messaging protocol specification for exchanging structured information in the implementation of web services in computer networks. SOAP can extend HTTP for XML messaging. SOAP provides data transport for Web services. SOAP can exchange complete documents or call a remote procedure. SOAP can be used for broadcasting a message.

[gRPC](#) is a modern, open source remote procedure call (RPC) framework developed by Google that can run anywhere. It enables client and server applications to communicate transparently, and makes it easier to build connected systems. It uses HTTP/2 for transport, Protocol Buffers as the interface description language, and provides features such as authentication, bidirectional streaming and flow control, blocking or nonblocking bindings, and cancellation and timeouts.

[LWM2M](#) is a protocol from the Open Mobile Alliance for M2M or IoT device management. Lightweight M2M enabler defines the application layer communication protocol between a LWM2M Server and a LWM2M Client, which is located in a LWM2M Device.

[Advanced Message Queuing Protocol \(AMQP\)](#) is an open standard for passing business messages between applications or organizations. It connects systems, feeds business processes with the information they need and reliably transmits onward the instructions that achieve their goals. The defining features of AMQP are message orientation, queuing, routing, reliability and security.

[Constrained Application Protocol \(CoAP\)](#) is a specialized web transfer protocol for use with constrained nodes and constrained networks in the Internet of Things. The protocol is designed for machine-to-machine (M2M) applications such as smart energy and building automation."

[Extensible Messaging and Presence Protocol \(XMPP\)](#) is a communication protocol for message-oriented middleware based on XML (Extensible Markup Language). It enables the near real-time exchange of structured yet extensible data between any two or more network entities.

[OASIS Message Queuing Telemetry Transport \(MQTT\)](#) is an open OASIS and ISO standard (ISO/IEC 20922) lightweight, publish-subscribe network protocol that transports messages between devices. The protocol usually runs over TCP/IP; however, any network protocol that provides ordered, lossless, bi-directional connections can support MQTT.

[Very Simple Control Protocol \(VSCP\)](#) is a free automation protocol suitable for all sorts of automation task where building- or home-automation is in the main focus. Its main advantage is that each VSCP-node can work completely autonomous, being part of distributed network of other nodes.

Operating systems (OS)

[Arm Mbed TLS](#) provides a comprehensive SSL/TLS solution and makes it easy for developers to include cryptographic and SSL/TLS capabilities in their software and embedded products. As an SSL library, it provides an intuitive API, readable source code and a minimal and highly configurable code footprint.

[Canonical Ubuntu Core](#) is a lean, strictly confined and fully transactional operating system. We designed it from the ground up, to focus on security and simplified maintenance, for appliances and large device networks. Ubuntu Core is powered by snaps - the universal Linux packaging format.

[Contiki-os](#) is an operating system for networked, memory-constrained systems with a focus on low-power wireless Internet of Things devices.

[Raspbian](#) is a Debian-based computer operating system for Raspberry Pi.

[RIOT](#) is the friendly OS for IoT

[RabbitMQ](#) is the most widely deployed open source message broker. With tens of thousands of users, RabbitMQ is one of the most popular open source message brokers. From T-Mobile to Runtastic, RabbitMQ is used worldwide at small startups and large enterprises. RabbitMQ is lightweight and easy to deploy on premises and in the cloud.

Middleware

[IoTSyS](#) is an integration middleware for the Internet of Things. It provides a communication stack for embedded devices based on IPv6, Web services, and OBIX to establish interoperable interfaces for smart objects.

[OpenIoT](#) is an open source middleware infrastructure will support flexible configuration and deployment of algorithms for collection, and filtering information streams stemming from the internet-connected objects, while at the same time generating and processing important business/applications events.

[OpenRemote](#) is an open source middleware project, which integrates many different protocols and solutions available for smart building, and smart city automation, and offers visualization tools.

[Kaa](#) is a Enterprise IoT Platform has been designed with heavy-duty, enterprise-grade IoT solutions in mind. It banishes a monolithic approach to architecture in favour of highly portable microservices, which allow for flexible rearrangement and customization even in the middle of the solution's lifecycle.

Node flow editors

[Node-RED](#) is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways. It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.

Toolkits

[KinomaJS](#) is a visual code editor designed to help developers build starter projects for Kinoma Create and Kinoma Element. The project is built on Angular 2(RC7) and runs in a web browser. The live version is hosted using Google App Engine, but you can modify and build it yourself by following the instructions in this document.

[IoT Toolkit](#) is a collection of libraries that enables communication with modern IoT based environments and devices. It is a high-performance collection of libraries optimized for minimum memory consumption in RAM, ROM, high speed, and versatility working on any device.

Data visualization

[Freeboard](#) is an open source real-time dashboard builder for IOT and other web mashups. A free open-source alternative to Geckoboard.

[ThingSpeak](#) is an IoT analytics platform service that allows you to aggregate, visualize, and analyze live data streams in the cloud. You can send data to ThingSpeak from your devices, create instant visualization of live data, and send alerts.

Search

[Thingful](#) is a Search Engine for the Internet of Things Find & use open IoT data from around the world.

Hardware

[Arduino Ethernet Shield 2](#) allows an Arduino board to connect to the internet using the Ethernet library and to read and write an SD card using the SD library. This shield is fully compatible with the former version, but relies on the newer W5500 chip.

[Raspberry Pi](#) is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote teaching of basic computer science in schools and in developing countries. Price range from \$10-45 depending on model.

[BeagleBone](#) is a low-power open-source single-board computer produced by Texas Instruments. It runs Android, Ubuntu and other Linux flavors.

[openPicus FlyportPro](#) is a system on a module dedicated to IoT and M2M application, especially for professional use. Following some details on the solution: Why FlyportPRO SoM? A system-on-module is the best solution for those customers looking for flexibility and for development time and risk reduction.

[Pinoccio](#) is a solution to add mesh networking capability and WiFi-Internet access to all your IoT devices, and it is Arduino compatible. Each board can assume the role of Scout in a Troop and one of the Scouts is the Lead to connect internet: Field Scouts talk to each other using a mesh network (called a Troop), using an extremely low-power radio.

In-memory data grids

[Ehcache](#) is an open source, standards-based cache that boosts performance, offloads your database, and simplifies scalability. It's the most widely-used Java-based cache because it's robust, proven, full-featured, and integrates with other popular libraries and frameworks.

[Hazelcast](#) is an open source in-memory data grid based on Java.

Home automation

[Home Assistant](#) is open source home automation that puts local control and privacy first. Powered by a worldwide community of tinkerers and DIY enthusiasts. Perfect to run on a Raspberry Pi or a local server.

[openHAB](#) is a cross-platform software with the aim to integrate all kinds of Smart Home technologies, devices, etc.

[Eclipse SmartHome](#) is a framework, not a ready-to-use solution. It offers a large set of features to choose from and leaves enough possibilities to design a Smart Home solution specific to your expectations. Its modular design brings millions of combinations and proves to be easily extensible by custom parts.

[The Thing System](#) is a set of software components and network protocols that aims to fix the Internet of Things. Our steward software is written in node.js making it both portable and easily extensible. It can run on your laptop, or fit onto a small single board computer like the Raspberry Pi.

Robotics

[Open Source Robotics Foundation](#) works with industry, academia, and government to create and support open software and hardware for use in robotics, from research and education to product development.

[ROS/ROS2](#) is a set of software libraries and tools that help you build robot applications. From drivers to state-of-the-art algorithms, and with powerful developer tools, ROS has what you need for your next robotics project.

Mesh networks

[Open Garden](#) develops the FireChat mobile application, which enables peer-to-peer mobile Internet connection sharing with faster and more efficient data transmissions by automatically and actively choosing and switching to the best available network without requiring users to manually sift through available networks to find the best one available.

[OpenWSN](#) is a project created at the University of California Berkeley and extended at the INRIA and at the Open University of Catalonia which aims to build an open standard-based and open source implementation of a complete constrained network protocol stack for wireless sensor networks and Internet of Things.

3. Open Source Security

[Back to the Top](#)



Open Source Security Guide

A guide covering Security including the applications, libraries and tools that will make you better and more efficient at securing your system operations and networks.

Resources

[Open Source Security Foundation \(OpenSSF\)](#) is a cross-industry collaboration that brings together leaders to improve the security of open source software by building a broader community, targeted initiatives, and best practices. The OpenSSF brings together open source security initiatives under one foundation to accelerate work through cross-industry support. Along with the Core Infrastructure Initiative and the Open Source Security Coalition, and will include new working groups that address vulnerability disclosures, security tooling and more.

[Microsoft Open Source Software Security](#)

[Cloudflare Open Source Security](#)

[The Seven Properties of Highly Secure Devices](#)

[How Layer 7 of the Internet Works](#)

[The 7 Kinds of Security](#)

[The Libgcrypt Reference Manual](#)

[The Open Web Application Security Project\(OWASP\) Foundation Top 10](#)

[Best Practices for Using Open Source Code from The Linux Foundation](#)

[AWS Certified Security - Specialty Certification](#)

[Microsoft Certified: Azure Security Engineer Associate](#)

[Google Cloud Certified Professional Cloud Security Engineer](#)

[Cisco Security Certifications](#)

[The Red Hat Certified Specialist in Security: Linux](#)

[Linux Professional Institute LPIC-3 Enterprise Security Certification](#)

[Cybersecurity Training and Courses from IBM Skills](#)

[Cybersecurity Courses and Certifications by Offensive Security](#)

[RSA Certification Program](#)

[Check Point Certified Security Expert\(CCSE\) Certification](#)

[Check Point Certified Security Administrator\(CCSA\) Certification](#)

[Check Point Certified Security Master \(CCSM\) Certification](#)

[Certified Cloud Security Professional\(CCSP\) Certification](#)

[Certified Information Systems Security Professional \(CISSP\) Certification](#)

[CCNP Routing and Switching](#)

[Certified Information Security Manager\(CISM\)](#)

[Wireshark Certified Network Analyst \(WCNA\)](#)

[Juniper Networks Certification Program Enterprise \(JNCP\)](#)

[Security Training Certifications and Courses from Udemy](#)

[Security Training Certifications and Courses from Coursera](#)

[Security Certifications Training from Pluarlsight](#)

Security Standards, Frameworks and Benchmarks

[STIGs Benchmarks - Security Technical Implementation Guides](#)

[CIS Benchmarks - CIS Center for Internet Security](#)

[NIST - Current FIPS](#)

[ISO Standards Catalogue](#)

Common Criteria for Information Technology Security Evaluation (CC) is an international standard (ISO / IEC 15408) for computer security. It allows an objective evaluation to validate that a particular product satisfies a defined set of security requirements.

ISO 22301 is the international standard that provides a best-practice framework for implementing an optimised BCMS (business continuity management system).

ISO27001 is the international standard that describes the requirements for an ISMS (information security management system). The framework is designed to help organizations manage their security practices in one place, consistently and cost-effectively.

ISO 27701 specifies the requirements for a PIMS (privacy information management system) based on the requirements of ISO 27001. It is extended by a set of privacy-specific requirements, control objectives and controls. Companies that have implemented ISO 27001 will be able to use ISO 27701 to extend their security efforts to cover privacy management.

EU GDPR (General Data Protection Regulation) is a privacy and data protection law that supersedes existing national data protection laws across the EU, bringing uniformity by introducing just one main data protection law for companies/organizations to comply with.

[CCPA \(California Consumer Privacy Act\)](#) is a data privacy law that took effect on January 1, 2020 in the State of California. It applies to businesses that collect California residents' personal information, and its privacy requirements are similar to those of the EU's GDPR (General Data Protection Regulation).

[Payment Card Industry \(PCI\) Data Security Standards \(DSS\)](#) is a global information security standard designed to prevent fraud through increased control of credit card data.

[SOC 2](#) is an auditing procedure that ensures your service providers securely manage your data to protect the interests of your company/organization and the privacy of their clients.

[NIST CSF](#) is a voluntary framework primarily intended for critical infrastructure organizations to manage and mitigate cybersecurity risk based on existing best practice.

Tools

[AppArmor](#) is an effective and easy-to-use Linux application security system. AppArmor proactively protects the operating system and applications from external or internal threats, even zero-day attacks, by enforcing good behavior and preventing both known and unknown application flaws from being exploited. AppArmor supplements the traditional Unix discretionary access control (DAC) model by providing mandatory access control (MAC). It has been included in the mainline Linux kernel since version 2.6.36 and its development has been supported by Canonical since 2009.

[SELinux](#) is a security enhancement to Linux which allows users and administrators more control over access control. Access can be constrained on such variables as which users and applications can access which resources. These resources may take the form of files. Standard Linux access controls, such as file modes (-rwxr-xr-x) are modifiable by the user and the applications which the user runs. Conversely, SELinux access controls are determined by a policy loaded on the system which may not be changed by careless users or misbehaving applications.

[Control Groups\(Cgroups\)](#) is a Linux kernel feature that allows you to allocate resources such as CPU time, system memory, network bandwidth, or any combination of these resources for user-defined groups of tasks (processes) running on a system.

[EarlyOOM](#) is a daemon for Linux that enables users to more quickly recover and regain control over their system in low-memory situations with heavy swap usage.

[Libgcrypt](#) is a general purpose cryptographic library originally based on code from GnuPG.

[Kali Linux](#) is an open source project that is maintained and funded by Offensive Security, a provider of world-class information security training and penetration testing services.

[Pi-hole](#) is a [DNS sinkhole](#) that protects your devices from unwanted content, without installing any client-side software, intended for use on a private network. It is designed for use on embedded devices with network capability, such as the Raspberry Pi, but it can be used on other machines running Linux and cloud implementations.

[Aircrack-ng](#) is a network software suite consisting of a detector, packet sniffer, WEP and WPA/WPA2-PSK cracker and analysis tool for 802.11 wireless LANs. It works with any wireless network interface controller whose driver supports raw monitoring mode and can sniff 802.11a, 802.11b and 802.11g traffic.

[Burp Suite](#) is a leading range of cybersecurity tools.

[KernelCI](#) is a community-based open source distributed test automation system focused on upstream kernel development. The primary goal of KernelCI is to use an open testing philosophy to ensure the quality, stability and long-term maintenance of the Linux kernel.

[Continuous Kernel Integration project](#) helps find bugs in kernel patches before they are committed to an upstream kernel tree. We are team of kernel developers, kernel testers, and automation engineers.

[eBPF](#) is a revolutionary technology that can run sandboxed programs in the Linux kernel without changing kernel source code or loading kernel modules. By making the Linux kernel programmable, infrastructure software can leverage existing layers, making them more intelligent and feature-rich without continuing to add additional layers of complexity to the system.

[Cilium](#) uses eBPF to accelerate getting data in and out of L7 proxies such as Envoy, enabling efficient visibility into API protocols like HTTP, gRPC, and Kafka.

[Hubble](#) is a Network, Service & Security Observability for Kubernetes using eBPF.

[Istio](#) is an open platform to connect, manage, and secure microservices. Istio's control plane provides an abstraction layer over the underlying cluster management platform, such as Kubernetes and Mesos.

[Certgen](#) is a convenience tool to generate and store certificates for Hubble Relay mTLS.

[Scapy](#) is a python-based interactive packet manipulation program & library.

[syzkaller](#) is an unsupervised, coverage-guided kernel fuzzer.

[SchedViz](#) is a tool for gathering and visualizing kernel scheduling traces on Linux machines.

[oss-fuzz](#) aims to make common open source software more secure and stable by combining modern fuzzing techniques with scalable, distributed execution.

[OSSEC](#) is a free, open-source host-based intrusion detection system. It performs log analysis, integrity checking, Windows registry monitoring, rootkit detection, time-based alerting, and active response.

[Metasploit Project](#) is a computer security project that provides information about security vulnerabilities and aids in penetration testing and IDS signature development.

[WFuzz](#) was created to facilitate the task in web applications assessments and it is based on a simple concept: it replaces any reference to the FUZZ keyword by the value of a given payload.

[Nmap](#) is a security scanner used to discover hosts and services on a computer network, thus building a "map" of the network.

[Patchwork](#) is a web-based patch tracking system designed to facilitate the contribution and management of contributions to an open-source project.

[pfSense](#) is a free and open source firewall and router that also features unified threat management, load balancing, multi WAN, and more.

[Snowpatch](#) is a continuous integration tool for projects using a patch-based, mailing-list-centric git workflow. This workflow is used by a number of well-known open source projects such as the Linux kernel.

[Snort](#) is an open-source, free and lightweight network intrusion detection system (NIDS) software for Linux and Windows to detect emerging threats.

[Wireshark](#) is a free and open-source packet analyzer. It is used for network troubleshooting, analysis, software and communications protocol development, and education.

[OpenSCAP](#) is U.S. standard maintained by [National Institute of Standards and Technology \(NIST\)](#). It provides multiple tools to assist administrators and auditors with assessment, measurement, and enforcement of security baselines. OpenSCAP maintains great flexibility and interoperability by reducing the costs of performing security audits. Whether you want to evaluate DISA STIGs, NIST's USGCB, or Red Hat's Security Response Team's content, all are supported by OpenSCAP.

[Tink](#) is a multi-language, cross-platform, open source library that provides cryptographic APIs that are secure, easy to use correctly, and harder to misuse.

[OWASP](#) is an online community, produces freely-available articles, methodologies, documentation, tools, and technologies in the field of web application security.

[Open Vulnerability and Assessment Language](#) is a community effort to standardize how to assess and report upon the machine state of computer systems. OVAL includes a language to encode system details, and community repositories of content. Tools and services that use OVAL provide enterprises with accurate, consistent, and actionable information to improve their security.

[ClamAV](#) is an open source antivirus engine for detecting trojans, viruses, malware & other malicious threats.

4. Linux

[Back to the Top](#)



Linux Guide

A guide covering Linux including it's multiple Desktop environments, Window managers, Tools, and Applications that will make you a better and more efficient Linux user.

Resources

[A Brief History of Linux](#)

[Contribute to FOSS](#) by Jason Evangelho, is an extensive Wiki page on how to contribute to the Linux and FOSS community.

[The Open Source Computer Science Degree by Forrest Knight](#)

[My Windows Subsystem for Linux\(WSL\) Guide](#)

A Cloud Guru Linux Training Certifications

Red Hat Training and Certification Program

Linux Foundation Training and Certification Program

[Linux Professional Institute\(LPI\)](#) is a non profit organization. Linux Professional Institute (LPI) is the global certification standard and career support organization for open source professionals. With more than 175,000 certification holders, it's the world's first and largest vendor-neutral Linux and open source certification body.

[Linux Foundation](#) is home to Linux, Node.js and other mission critical projects that form the backbone of modern internet services, including Pinterest. Joining The Linux Foundation is a great way for established companies like ours to support those communities.

[Cloud Native Computing Foundation \(CNCF\)](#) hosts critical components of the global technology infrastructure. CNCF brings together the world's top developers, end users, and vendors and runs the largest open source developer conferences. CNCF is part of the nonprofit Linux Foundation.

[RISC-V Foundation](#) is a non-profit corporation controlled by its members to drive forward the adoption and implementation of the free and open RISC-V instruction set architecture (ISA).

[Open Source Security Foundation \(OpenSSF\)](#) is a cross-industry collaboration that brings together leaders to improve the security of open source software by building a broader community, targeted initiatives, and best practices. The OpenSSF brings together open source security initiatives under one foundation to accelerate work through cross-industry support. Along with the Core Infrastructure Initiative and the Open Source Security Coalition, and will include new working groups that address vulnerability disclosures, security tooling and more.

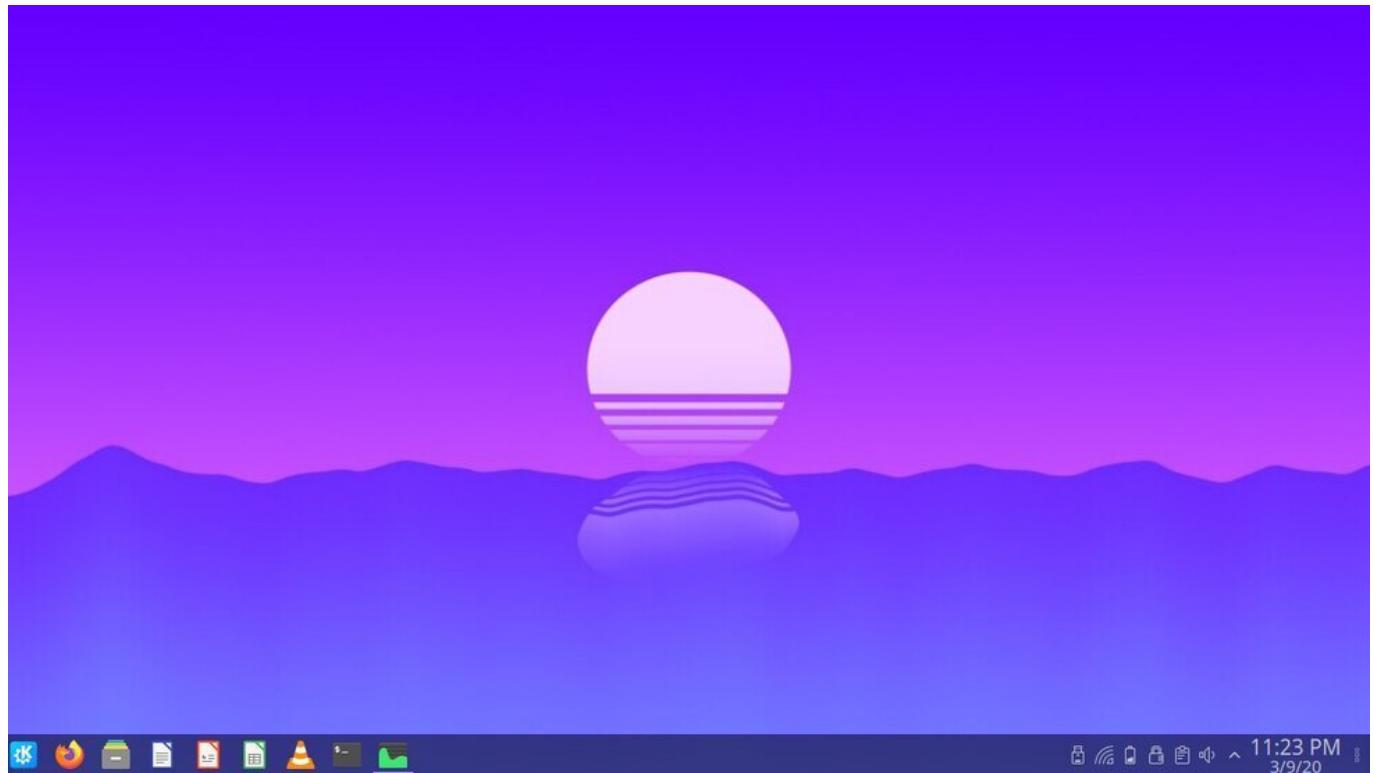
[Free Software Foundation \(FSF\)](#) is a nonprofit with a worldwide mission to promote computer user freedom.

Desktop Environments



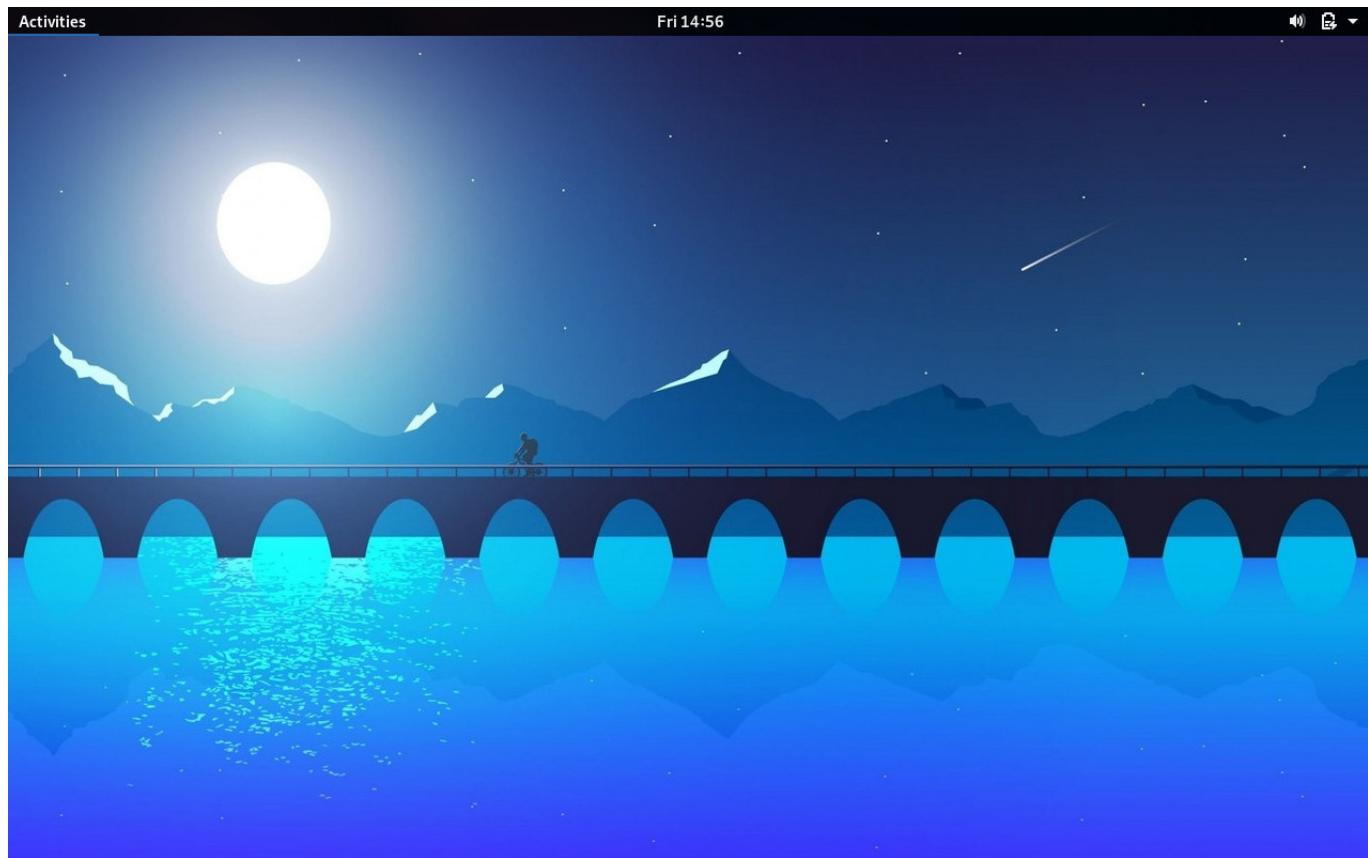
KDE Plasma Desktop

[KDE Plasma Desktop](#), is a beautiful Windows-like desktop that you use to surf the web, keep in touch with colleagues, friends and family, manage your files, enjoy music and videos; and get creative and productive at work. The KDE community develops and maintains more than 200 applications which run on any Linux desktop, and often other platforms too.



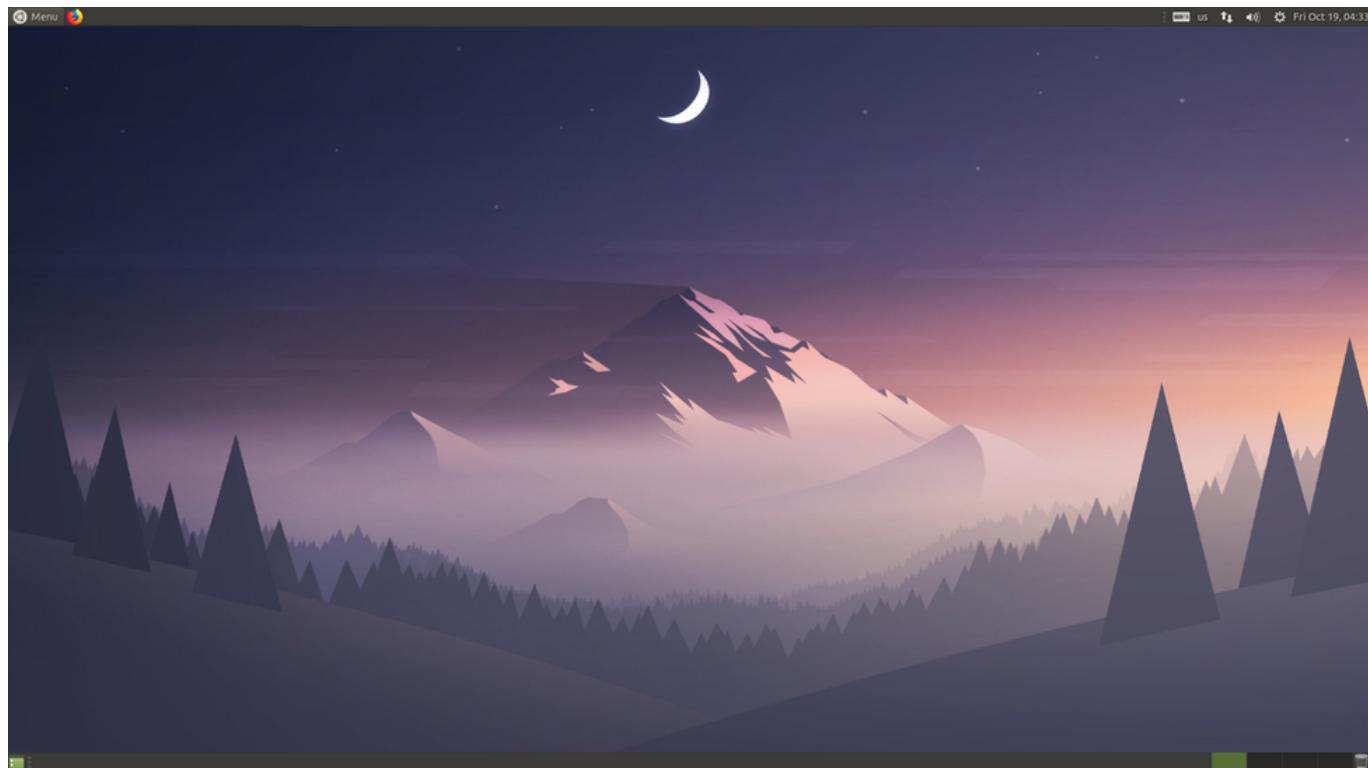
GNOME Desktop

[GNOME Desktop](#), An easy and elegant way to use your computer, GNOME 3 is designed to put you in control and get things done.



MATE Desktop

[MATE Desktop](#), is the continuation of GNOME 2 desktop environment.



Budgie Desktop

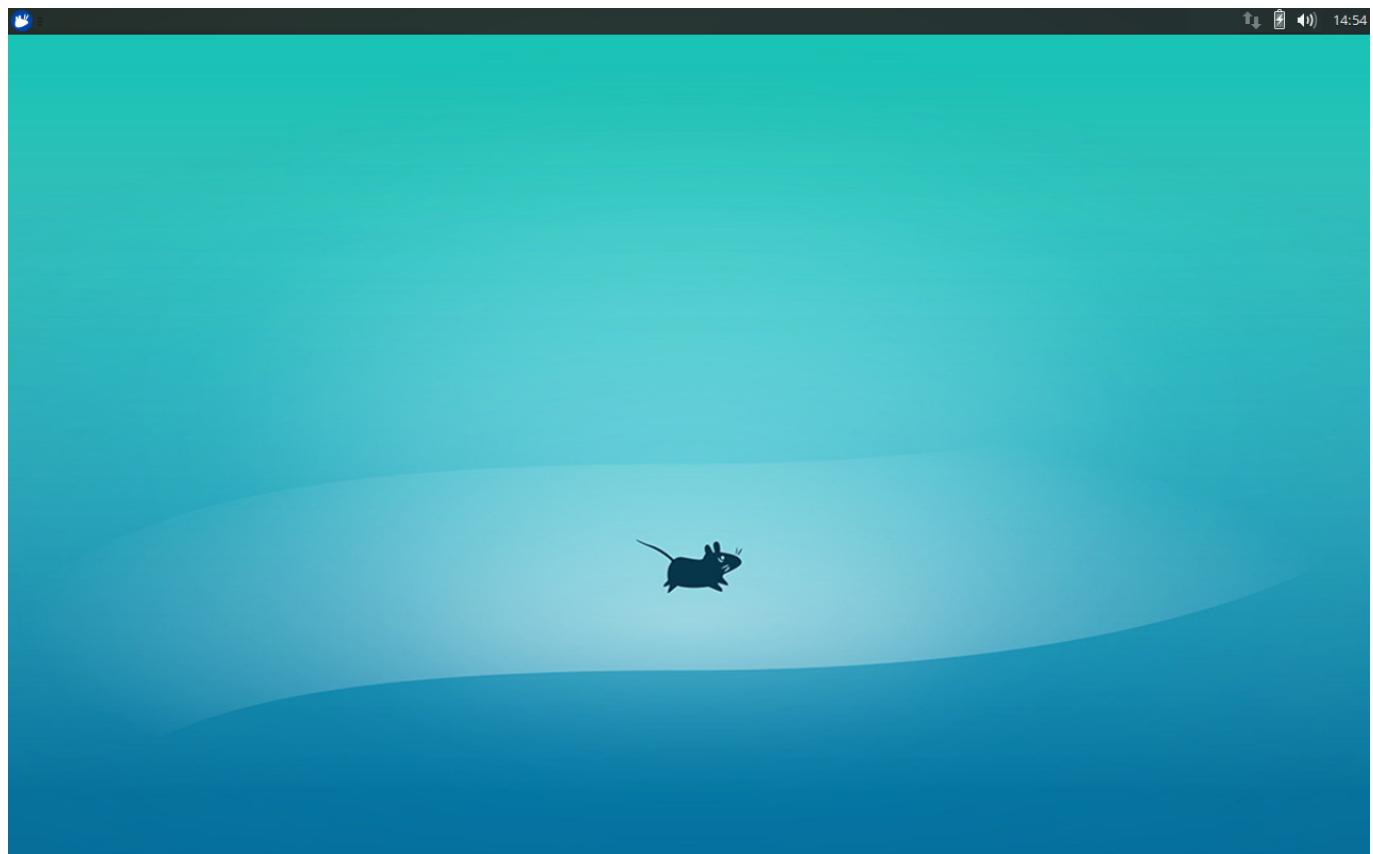
[Budgie Desktop](#), a feature-rich, modern desktop designed to keep out the way of the user.





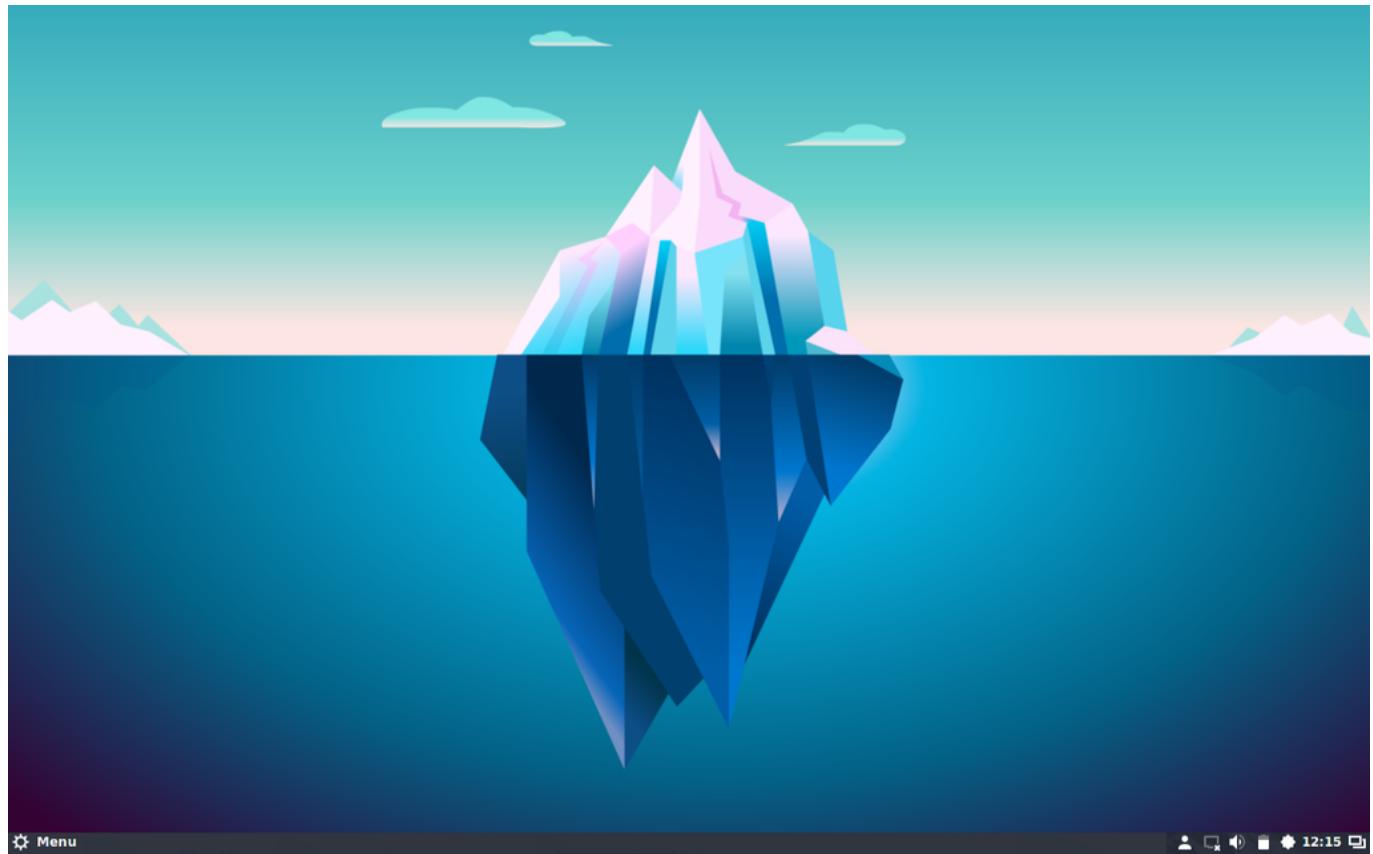
XFCE Desktop

[XFCE Desktop](#), a lightweight desktop environment for UNIX-like operating systems.



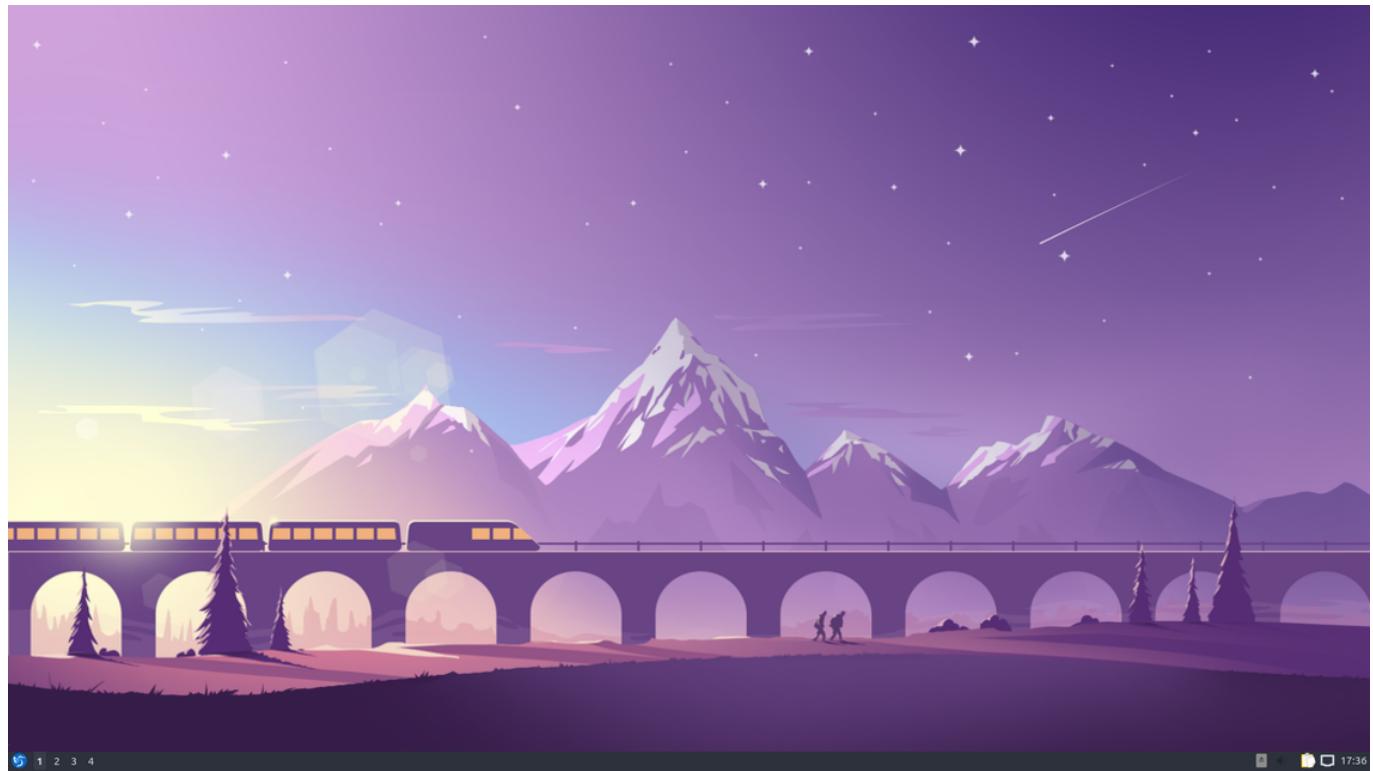
Cinnamon Desktop

[Cinnamon Desktop](#), a Linux desktop which provides advanced innovative features and a traditional user experience.



LXQt Desktop

[LXQt Desktop](#), the Lightweight Qt Desktop Environment.



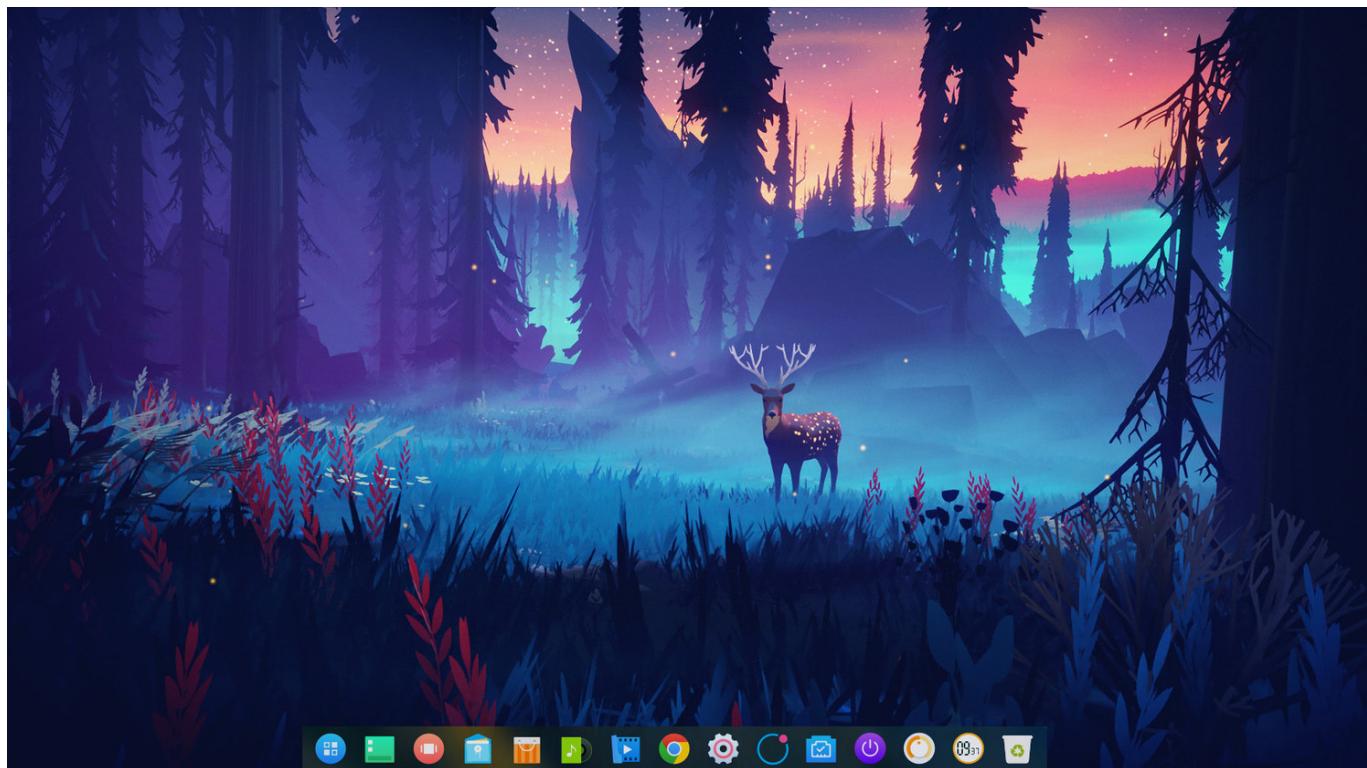
Pantheon Desktop

[Pantheon Desktop](#), a GTK desktop based on the GNOME software base maintained by the elementary OS developers.



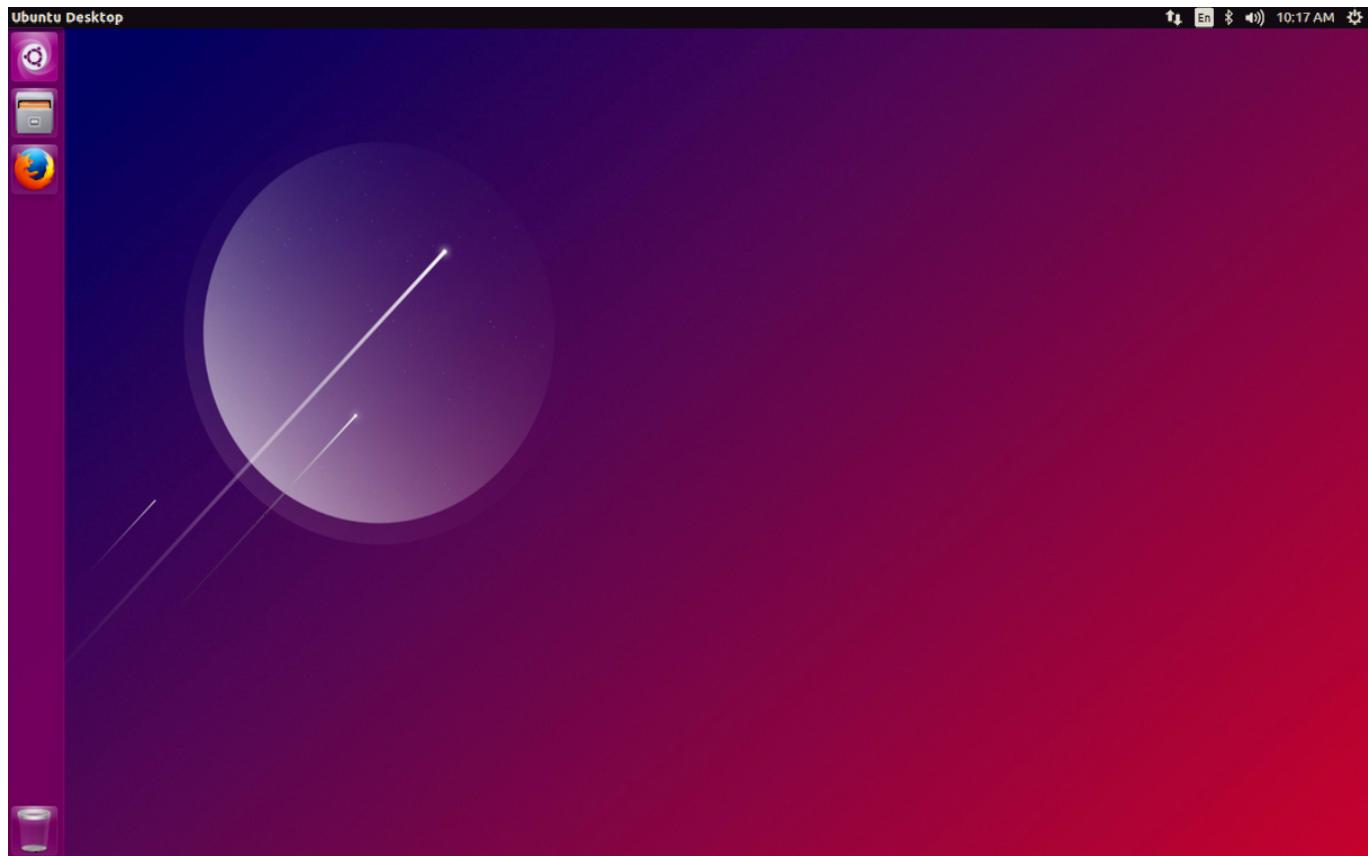
Deepin Desktop

[Deepin Desktop](#), an elegant, easy to use and reliable domestic desktop environment.



Unity Desktop

[Unity Desktop](#), a graphical shell for the GNOME desktop environment originally developed by Canonical Ltd.

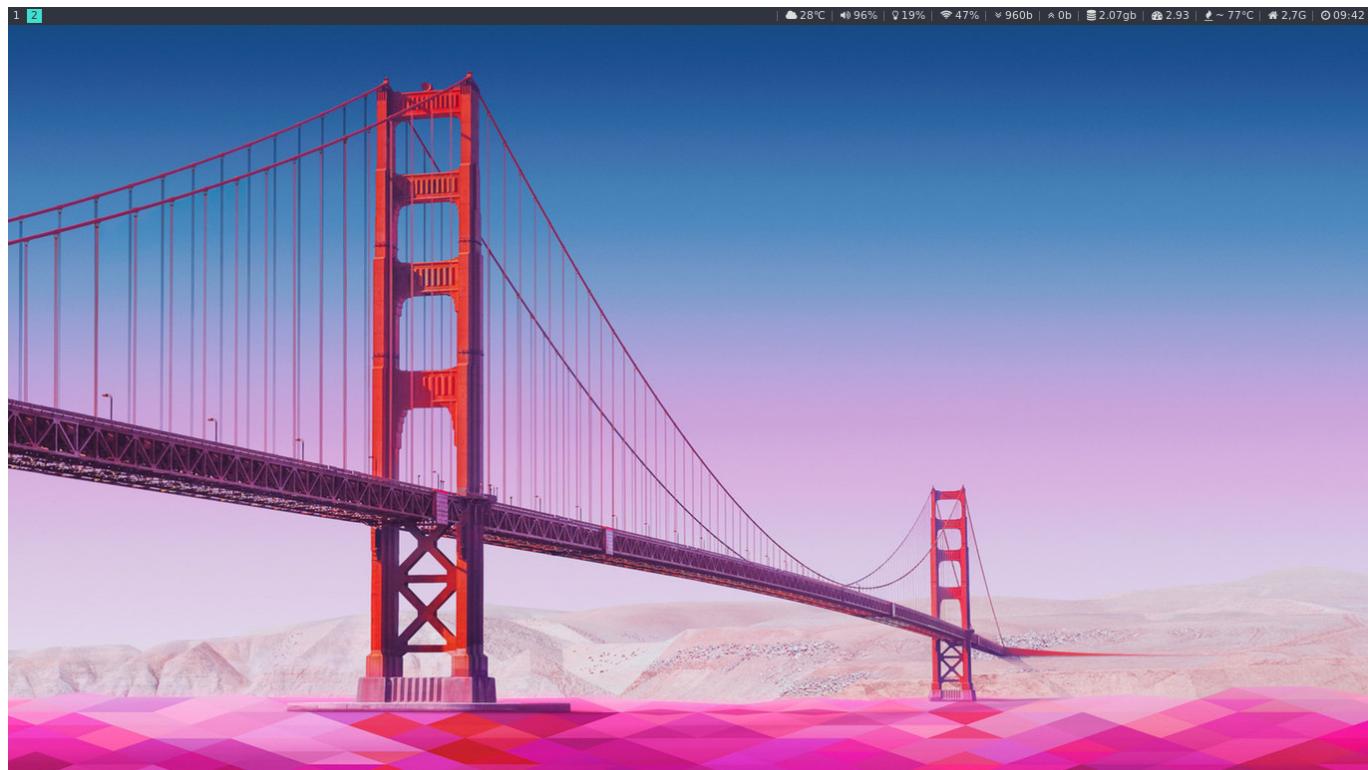


Windows mangers



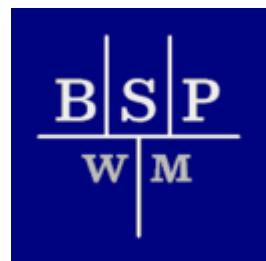
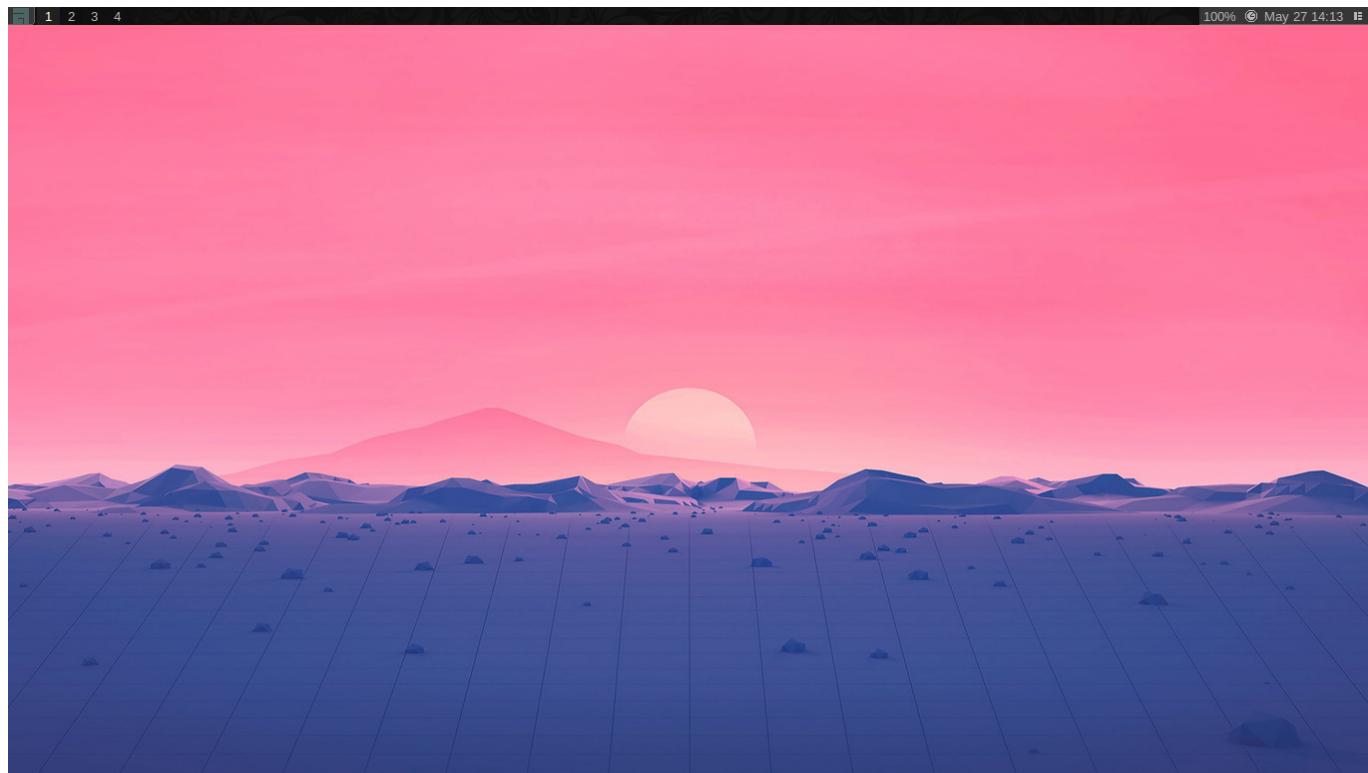
i3 wm

[i3-wm](#), a tiling window manager for X11.



awesome **wm**

[Awesome](#), a highly configurable, next generation framework window manager for X.



Bspwm

[Bspwm](#), a tiling window manager based on binary space partitioning with a focus on resource efficiency.



Linux Companies

[Red Hat](#) makers of Red Hat Enterprise Linux and sponsors the [Fedora Project](#). Red Hat also develops [CentOS Stream](#), which is a continuously delivered distro that tracks just ahead of Red Hat Enterprise Linux (RHEL) development, positioned as a midstream between Fedora Linux and RHEL.

[Canonical](#) makers of Ubuntu Desktop and Server.

[SUSE](#) makers of SUSE Enterprise Linux.

[Manjaro](#) makers of Manjaro Linux.

[System76](#) makers of [Pop!_OS](#)

Linux Hardware Vendors

[Dell](#)

[Lenovo](#)

[HP](#)

[System76](#)

[Tuexdo Computers](#)

[Entroware](#)

[Pine64](#)

[Purism](#)

Linux in the Cloud

[Amazon Linux 2](#) a next generation of Amazon Linux, a Linux server operating system from Amazon Web Services (AWS). It provides a secure, stable, and high performance execution environment to develop and run cloud and enterprise applications. With Amazon Linux 2, you get an application environment that offers long term support with access to the latest innovations in the Linux ecosystem.

[Bottlerocket](#) is a Linux-based open-source operating system that is purpose-built by Amazon Web Services for running containers on virtual machines or bare metal hosts. Most customers today run containerized applications on general-purpose operating systems that are updated package-by-package, which makes OS updates difficult to automate.

[Getting Started with Bottlerocket and Certified AWS Partners](#)

[Using Bottlerocket with Amazon ECS](#)

[Amazon Linux AMI](#) is a supported and maintained Linux image provided by Amazon Web Services for use on Amazon Elastic Compute Cloud (Amazon EC2). It is designed to provide a stable, secure, and high performance execution environment for applications running on Amazon EC2.

[Ubuntu on AWS](#)

[Ubuntu on Azure](#)

[Ubuntu Pro for Azure](#)

[Linux on Azure](#)

[Creating a Linux VM in the Azure portal](#)

Azure Sphere is a secured, high-level application platform with built-in communication and security features for internet-connected devices. The platform consists of the integration of hardware built around a secured silicon chip; the Azure Sphere OS, a custom high-level Linux-based operating system; and the Azure Sphere Security Service, a cloud-based security service that provides continuous security.

[Creating a virtual Linux workstation in Google Cloud.](#)

[Preparing Linux VMs migrating to Google Cloud with Migrate for Compute Engine](#)

[Understanding OpenStack on Red Hat](#)

[Understanding OpenStack on Ubuntu](#)

[Initial Server Setup with Ubuntu 20.04 on DigitalOcean](#)

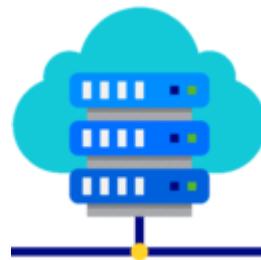
[Tutorials for Setting up Linux workspaces on DigitalOcean](#)

[DigitalOcean Marketplace](#)

[Linode](#)

5. Networking

[Back to the Top](#)



Networking Guide

A Networking Guide for getting started with basic/core network components such as wired and wireless network design, configuration, hardware, protocols, security, backup, recovery, and virtualization.

Resources

[AWS Certified Security - Specialty Certification](#)

[Microsoft Certified: Azure Security Engineer Associate](#)

[Google Cloud Certified Professional Cloud Security Engineer](#)

[Cisco Security Certifications](#)

[The Red Hat Certified Specialist in Security: Linux](#)

[Linux Professional Institute LPIC-3 Enterprise Security Certification](#)

[Cybersecurity Training and Courses from IBM Skills](#)

[Cybersecurity Courses and Certifications by Offensive Security](#)

[Citrix Certified Associate – Networking\(CCA-N\)](#)

[Citrix Certified Professional – Virtualization\(CCP-V\)](#)

[CCNP Routing and Switching](#)

[Certified Information Security Manager\(CISM\)](#)

[Wireshark Certified Network Analyst \(WCNA\)](#)

[Juniper Networks Certification Program Enterprise \(JNCP\)](#)

[Networking courses and specializations from Coursera](#)

[Network & Security Courses from Udemy](#)

[Network & Security Courses from edX](#)

Tools & Networking Concepts

- Connection: In networking, a connection refers to pieces of related information that are transferred through a network. This generally infers that a connection is built before the data transfer (by following the procedures laid out in a protocol) and then is deconstructed at the end of the data transfer.
- Packet: A packet is, generally speaking, the most basic unit that is transferred over a network. When communicating over a network, packets are the envelopes that carry your data (in pieces) from one end point to the other.

Packets have a header portion that contains information about the packet including the source and destination, timestamps, network hops. The main portion of a packet contains the actual data being transferred. It is sometimes called the body or the payload.

- Network Interface: A network interface can refer to any kind of software interface to networking hardware. For instance, if you have two network cards in your computer, you can control and configure each network interface associated with them individually.

A network interface may be associated with a physical device, or it may be a representation of a virtual interface. The "loop-back" device, which is a virtual interface to the local machine, is an example of this.

- LAN: LAN stands for "local area network". It refers to a network or a portion of a network that is not publicly accessible to the greater internet. A home or office network is an example of a LAN.
- WAN: WAN stands for "wide area network". It means a network that is much more extensive than a LAN. While WAN is the relevant term to use to describe large, dispersed networks in general, it is usually meant to mean the internet, as a whole.

If an interface is connected to the WAN, it is generally assumed that it is reachable through the internet.

- Protocol: A protocol is a set of rules and standards that basically define a language that devices can use to communicate. There are a great number of protocols in use extensively in networking, and they are often implemented in different layers.

Some low level protocols are TCP, UDP, IP, and ICMP. Some familiar examples of application layer protocols, built on these lower protocols, are HTTP (for accessing web content), SSH, TLS/SSL, and FTP.

- Port: A port is an address on a single machine that can be tied to a specific piece of software. It is not a physical interface or location, but it allows your server to be able to communicate using more than one application.
- Firewall: A firewall is a program that decides whether traffic coming into a server or going out should be allowed. A firewall usually works by creating rules for which type of traffic is acceptable on which ports. Generally, firewalls block ports that are not used by a specific application on a server.
- NAT: Network address translation is a way to translate requests that are incoming into a routing server to the relevant devices or servers that it knows about in the LAN. This is usually implemented in physical LANs as a way to route requests through one IP address to the necessary backend servers.
- VPN: Virtual private network is a means of connecting separate LANs through the internet, while maintaining privacy. This is used as a means of connecting remote systems as if they were on a local network, often for security reasons.

Network Layers

While networking is often discussed in terms of topology in a horizontal way, between hosts, its implementation is layered in a vertical fashion throughout a computer or network. This means is that there are multiple technologies and protocols that are built on top of each other in order for communication to function more easily. Each successive, higher layer abstracts the raw data a little bit more, and makes it simpler to use for applications and users. It also allows you to leverage lower layers in new ways without having to invest the time and energy to develop the protocols and applications that handle those types of traffic.

As data is sent out of one machine, it begins at the top of the stack and filters downwards. At the lowest level, actual transmission to another machine takes place. At this point, the data travels back up through the layers of the other computer. Each layer has the ability to add its own "wrapper" around the data that it receives from the adjacent layer, which will help the layers that come after decide what to do with the data when it is passed off.

One method of talking about the different layers of network communication is the OSI model. OSI stands for Open Systems Interconnect. This model defines seven separate layers. The layers in this model are:

- Application: The application layer is the layer that the users and user-applications most often interact with. Network communication is discussed in terms of availability of resources, partners to communicate with, and data synchronization.
- Presentation: The presentation layer is responsible for mapping resources and creating context. It is used to translate lower level networking data into data that applications expect to see.
- Session: The session layer is a connection handler. It creates, maintains, and destroys connections between nodes in a persistent way.
- Transport: The transport layer is responsible for handing the layers above it a reliable connection. In this context, reliable refers to the ability to verify that a piece of data was received intact at the other end of the connection. This layer can resend information that has been dropped or corrupted and can acknowledge the receipt of data to remote computers.
- Network: The network layer is used to route data between different nodes on the network. It uses addresses to be able to tell which computer to send information to. This layer can also break apart larger messages into smaller chunks to be reassembled on the opposite end.
- Data Link: This layer is implemented as a method of establishing and maintaining reliable links between different nodes or devices on a network using existing physical connections.
- Physical: The physical layer is responsible for handling the actual physical devices that are used to make a connection. This layer involves

the bare software that manages physical connections as well as the hardware itself (like Ethernet).

The TCP/IP model, more commonly known as the Internet protocol suite, is another layering model that is simpler and has been widely adopted. It defines the four separate layers, some of which overlap with the OSI model:

- Application: In this model, the application layer is responsible for creating and transmitting user data between applications. The applications can be on remote systems, and should appear to operate as if locally to the end user.

The communication takes place between peers network.

- Transport: The transport layer is responsible for communication between processes. This level of networking utilizes ports to address different services. It can build up unreliable or reliable connections depending on the type of protocol used.
- Internet: The internet layer is used to transport data from node to node in a network. This layer is aware of the endpoints of the connections, but does not worry about the actual connection needed to get from one place to another. IP addresses are defined in this layer as a way of reaching remote systems in an addressable manner.
- Link: The link layer implements the actual topology of the local network that allows the internet layer to present an addressable interface. It establishes connections between neighboring nodes to send data.

Interfaces

Interfaces are networking communication points for your computer. Each interface is associated with a physical or virtual networking device. Typically, your server will have one configurable network interface for each Ethernet or wireless internet card you have. In addition, it will define a virtual network interface called the "loopback" or localhost interface. This is used as an interface to connect applications and processes on a single computer to other applications and processes. You can see this referenced as the "lo" interface in many tools.

Protocols

Networking works by piggybacks on a number of different protocols on top of each other. In this way, one piece of data can be transmitted using multiple protocols encapsulated within one another.

Media access control is a communications protocol that is used to distinguish specific devices. Each device is supposed to get a unique MAC address during the manufacturing process that differentiates it from every other device on the internet. Addressing hardware by the MAC address allows you to reference a

device by a unique value even when the software on top may change the name for that specific device during operation. Media access control is one of the only protocols from the link layer that you are likely to interact with on a regular basis.

The IP protocol is one of the fundamental protocols that allow the internet to work. IP addresses are unique on each network and they allow machines to address each other across a network. It is implemented on the internet layer in the IP/TCP model. Networks can be linked together, but traffic must be routed when crossing network boundaries. This protocol assumes an unreliable network and multiple paths to the same destination that it can dynamically change between. There are a number of different implementations of the protocol. The most common implementation today is IPv4, although IPv6 is growing in popularity as an alternative due to the scarcity of IPv4 addresses available and improvements in the protocols capabilities.

ICMP: internet control message protocol is used to send messages between devices to indicate the availability or error conditions. These packets are used in a variety of network diagnostic tools, such as ping and traceroute. Usually ICMP packets are transmitted when a packet of a different kind meets some kind of a problem. Basically, they are used as a feedback mechanism for network communications.

TCP: Transmission control protocol is implemented in the transport layer of the IP/TCP model and is used to establish reliable connections. TCP is one of the protocols that encapsulates data into packets. It then transfers these to the remote end of the connection using the methods available on the lower layers. On the other end, it can check for errors, request certain pieces to be resent, and reassemble the information into one logical piece to send to the application layer. The protocol builds up a connection prior to data transfer using a system called a three-way handshake. This is a way for the two ends of the communication to acknowledge the request and agree upon a method of ensuring data reliability. After the data has been sent, the connection is torn down using a similar four-way handshake. TCP is the protocol of choice for many of the most popular uses for the internet, including WWW, FTP, SSH, and email. It is safe to say that the internet we know today would not be here without TCP.

UDP: User datagram protocol is a popular companion protocol to TCP and is also implemented in the transport layer. The fundamental difference between UDP and TCP is that UDP offers unreliable data transfer. It does not verify that data has been received on the other end of the connection. This might sound like a bad thing, and for many purposes, it is. However, it is also extremely important for some functions. It's not required to wait for confirmation that the data was received and forced to resend data, UDP is much faster than TCP. It does not establish a connection with the remote host, it simply fires off the data to that host and doesn't care if it is accepted or not. Since UDP is a simple transaction, it is useful for simple communications like querying for network resources. It also doesn't maintain a state, which makes it great for transmitting data from one machine to many real-time clients. This makes it ideal for VOIP, games, and other applications that cannot afford delays.

HTTP: Hypertext transfer protocol is a protocol defined in the application layer that forms the basis for communication on the web. HTTP defines a number of functions that tell the remote system what you are requesting. For instance, GET, POST, and DELETE all interact with the requested data in a different way.

[JSON Web Token \(JWT\)](#) is a compact URL-safe means of representing claims to be transferred between two parties. The claims in a JWT are encoded as a JSON object that is digitally signed using JSON Web Signature (JWS).

[OAuth 2.0](#) is an open source authorization framework that enables applications to obtain limited access to user accounts on an HTTP service, such as Amazon, Google, Facebook, Microsoft, Twitter GitHub, and DigitalOcean. It works by delegating user authentication to the service that hosts the user account, and authorizing third-party applications to access the user account.

[FTP](#): File transfer protocol is in the application layer and provides a way of transferring complete files from one host to another. It is inherently insecure, so it is not recommended for any externally facing network unless it is implemented as a public, download-only resource.

[DNS](#): Domain name system is an application layer protocol used to provide a human-friendly naming mechanism for internet resources. It is what ties a domain name to an IP address and allows you to access sites by name in your browser.

[SSH](#): Secure shell is an encrypted protocol implemented in the application layer that can be used to communicate with a remote server in a secure way. Many additional technologies are built around this protocol because of its end-to-end encryption and ubiquity. There are many other protocols that we haven't covered that are equally important. However, this should give you a good overview of some of the fundamental technologies that make the internet and networking possible.

Virtualization

[KVM \(for Kernel-based Virtual Machine\)](#) is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (Intel VT or AMD-V). It consists of a loadable kernel module, kvm.ko, that provides the core virtualization infrastructure and a processor specific module, kvm-intel.ko or kvm-amd.ko.

[QEMU](#) is a fast processor emulator using a portable dynamic translator. QEMU emulates a full system, including a processor and various peripherals. It can be used to launch a different Operating System without rebooting the PC or to debug system code.

[Hyper-V](#) enables running virtualized computer systems on top of a physical host. These virtualized systems can be used and managed just as if they were physical computer systems, however they exist in virtualized and isolated environment. Special software called a hypervisor manages access between the virtual systems and the physical hardware resources. Virtualization enables quick deployment of computer systems, a way to quickly restore systems to a previously known good state, and the ability to migrate systems between physical hosts.

[VirtManager](#) is a graphical tool for managing virtual machines via libvirt. Most usage is with QEMU/KVM virtual machines, but Xen and libvirt LXC containers are well supported. Common operations for any libvirt driver should work.

[oVirt](#) is an open-source distributed virtualization solution, designed to manage your entire enterprise infrastructure. oVirt uses the trusted KVM hypervisor and is built upon several other community projects, including libvirt, Gluster, PatternFly, and Ansible. Founded by Red Hat as a community project on which Red Hat Enterprise Virtualization is based allowing for centralized management of virtual machines, compute, storage and networking resources, from an easy-to-use web-based front-end with platform independent access.

[Xen](#) is focused on advancing virtualization in a number of different commercial and open source applications, including server virtualization, Infrastructure as a Services (IaaS), desktop virtualization,

security applications, embedded and hardware appliances, and automotive/aviation.

[Ganeti](#) is a virtual machine cluster management tool built on top of existing virtualization technologies such as Xen or KVM and other open source software. Once installed, the tool assumes management of the virtual instances (Xen DomU).

[Packer](#) is an open source tool for creating identical machine images for multiple platforms from a single source configuration. Packer is lightweight, runs on every major operating system, and is highly performant, creating machine images for multiple platforms in parallel. Packer does not replace configuration management like Chef or Puppet. In fact, when building images, Packer is able to use tools like Chef or Puppet to install software onto the image.

[Vagrant](#) is a tool for building and managing virtual machine environments in a single workflow. With an easy-to-use workflow and focus on automation, Vagrant lowers development environment setup time, increases production parity, and makes the "works on my machine" excuse a relic of the past. It provides easy to configure, reproducible, and portable work environments built on top of industry-standard technology and controlled by a single consistent workflow to help maximize the productivity and flexibility of you and your team.

[VMware Workstation](#) is a hosted hypervisor that runs on x64 versions of Windows and Linux operating systems; it enables users to set up virtual machines on a single physical machine, and use them simultaneously along with the actual machine.

[VirtualBox](#) is a powerful x86 and AMD64/Intel64 virtualization product for enterprise as well as home use. Not only is VirtualBox an extremely feature rich, high performance product for enterprise customers.

6. Cloud-Native

[Back to the Top](#)



Cloud Native Guide

A guide for using Cloud Native tools and building applications.

Resources

[CNCF Cloud Native Interactive Landscape](#)

[Build Cloud-Native applications in Microsoft Azure](#)

[Cloud-Native application development for Google Cloud](#)

[Cloud-Native development for Amazon Web Services](#)

[Cloud Native Computing Foundation Training and Certification Program](#)

[Cloud Foundry Developer Training and Certification Program](#)

[Cloud-Native Architecture Course on Pluralsight](#)

[AWS Fundamentals: Going Cloud-Native on Coursera](#)

[Developing Cloud-Native Apps w/ Microservices Architectures course on Udemy](#)

[How load balancing works for cloud native applications with Azure Application Gateway on LinkedIn Learning](#)

[Developing Cloud Native Applications course on edX](#)

[Cloud Native courses from IBM](#)

DevOps

Application Framework

[Spring Boot](#) is an open-source micro framework maintained by Pivotal, which was acquired by VMware in 2019. It provides Java developers with a platform to get started with an auto configurable production-grade Spring application.

[Apache Mesos](#) is a cluster manager that provides efficient resource isolation and sharing across distributed applications, or frameworks. It can run Hadoop, Jenkins, Spark, Aurora, and other frameworks on a dynamically shared pool of nodes.

[Apache Spark](#) is a unified analytics engine for big data processing, with built-in modules for streaming, SQL, machine learning and graph processing.

[Apache Hadoop](#) is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures.

Runtime Platform

[BOSH](#) is a tool that prepares your infrastructure for what needs to be managed. BOSH espouses software engineering best practices, such as continuous delivery, by making it easy to create software releases that automatically update complex distributed systems with simple commands. Due to the flexibility and power of BOSH, Google and VMware made it the heart of the Kubo project, now called the Cloud Foundry Container Runtime, based on Kubernetes.

Infrastructure Automation

[Maven](#) is a build automation tool used primarily for Java projects. Maven can also be used to build and manage projects written in C#, Ruby, Scala, and other languages. The Maven project is hosted by the Apache Software Foundation.

[Gradle](#) is an open-source build-automation system that builds upon the concepts of Apache Ant and Apache Maven and introduces a Groovy-based domain-specific language instead of the XML form used by Apache Maven for declaring the project configuration.

[Chef](#) is an effortless Infrastructure Suite offers visibility into security and compliance status across all infrastructure and makes it easy to detect and correct issues long before they reach production.

[Puppet](#) is an open source tool that makes continuous integration and delivery of your software on traditional or containerized infrastructure easy by pulling together all your existing tools and giving you flexibility to deploy your way.

[Ansible](#) is an open-source software provisioning, configuration management, and application-deployment tool. It runs on many Unix-like systems, and can configure both Unix-like systems as well as Microsoft Windows.

[Salt](#) is Python-based, open-source software for event-driven IT automation, remote task execution, and configuration management. Supporting the "Infrastructure as Code" approach to data center system and network deployment and management, configuration automation, SecOps orchestration, vulnerability remediation, and hybrid cloud control.

[Terraform](#) is an open-source infrastructure as code software tool created by HashiCorp. It enables users to define and provision a datacenter infrastructure using a high-level configuration language known as Hashicorp Configuration Language (HCL), or optionally JSON.

Cloud Infrastructure

[Amazon web service\(AWS\)](#) is a platform that offers flexible, reliable, scalable, easy-to-use and cost-effective cloud computing solutions. The AWS platform is developed with a combination of infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS) offerings.

[Microsoft Azure](#) is a cloud computing service created by Microsoft for building, testing, deploying, and managing applications and services through Microsoft-managed data centers.

[Azure DevOps](#) is a set of services for teams to share code, track work, and ship software; CLIs Build, deploy, diagnose, and manage multi-platform, scalable apps and services; Azure Pipelines Continuously build, test, and deploy to any platform and cloud; Azure Lab Services Set up labs for classrooms, trials, development and testing, and other scenarios.

[Azure Draft](#) is a tool for developers to create cloud-native applications on Kubernetes.

[Google Cloud Platform](#) integrates industry-leading tools(data management, hybrid & multi-cloud, and AI & ML) with Cloud Storage for enhanced support with everything from security and data transfer, to data backup and archive. Expand all . Backup, archival, and disaster recovery. Along with File systems and gateways.

[OpenStack](#) is a free and open-source software platform for cloud computing, mostly deployed as infrastructure-as-a-service that controls large pools of compute, storage, and networking resources throughout a datacenter, managed through a dashboard or via the OpenStack API. OpenStack works with popular enterprise and open source technologies making it ideal for heterogeneous infrastructure.

[Cloud Foundry](#) is an open source, multi cloud application platform as a service that makes it faster and easier to build, test, deploy and scale applications, providing a choice of clouds, developer frameworks, and application services. It is an open source project and is available through a variety of private cloud distributions and public cloud instances.

Continuous Integration/Continuous Delivery

[Bamboo](#) is a continuous integration (CI) server that can be used to automate the release management for a software application, creating a continuous delivery pipeline.

[Drone](#) is a Continuous Delivery system built on container technology. Drone uses a simple YAML configuration file, a superset of docker-compose, to define and execute Pipelines inside Docker containers.

[Travis CI](#) is a hosted continuous integration service used to build and test software projects hosted at GitHub.

[Circle CI](#) is a continuous integration and continuous delivery platform that helps software teams work smarter, faster.

[Team City](#) is a build management and continuous integration server from JetBrains.

[Shipable](#) simplifies DevOps and makes it systematic with an Assembly Line platform that is heterogeneous, flexible, and provides complete visibility across your DevOps workflows.

[Spinnaker](#) is an open source, multi-cloud continuous delivery platform for releasing software changes with high velocity and confidence.

[Prow](#) is a Kubernetes based CI/CD system. Jobs can be triggered by various types of events and report their status to many different services. In addition to job execution, Prow provides GitHub automation in the form of policy enforcement, chat-ops via /foo style commands, and automatic PR merging. Prow has a microservice architecture implemented as a collection of container images that run as Kubernetes deployments.

Microservices

[AWS ECS](#) is a highly scalable, high-performance container orchestration service that supports Docker containers and allows you to easily run and scale containerized applications on AWS. Amazon ECS eliminates the need for you to install and operate your own container orchestration software, manage and scale a cluster of virtual machines, or schedule containers on those virtual machines.

[AWS CodeBuild](#) is a fully managed continuous integration service that compiles source code, runs tests, and produces software packages that are ready to deploy. With CodeBuild, you don't need to provision, manage, and scale your own build servers.

[CFEngine](#) is an open-source configuration management system, written by Mark Burgess. Its primary function is to provide automated configuration and maintenance of large-scale computer systems, including the unified management of servers, desktops, consumer and industrial devices, embedded networked devices, mobile smartphones, and tablet computers.

[Octpus Deploy](#) is the deployment automation server for your entire team, designed to make it easy to orchestrate releases and deploy applications, whether on-premises or in the cloud.

[AWS CodeDeploy](#) is a fully managed deployment service that automates software deployments to a variety of compute services such as Amazon EC2, AWS Fargate, AWS Lambda, and your on-premises servers. AWS CodeDeploy makes it easier for you to rapidly release new features, helps you avoid downtime during application deployment, and handles the complexity of updating your applications.

[AWS Lambda](#) is an event-driven, serverless computing platform provided by Amazon as a part of the Amazon Web Services. It is a computing service that runs code in response to events and automatically manages the computing resources required by that code.

[Traefik](#) is an open-source Edge Router that makes publishing your services a fun and easy experience. It receives requests on behalf of your system and finds out which components are responsible for handling them. What sets Traefik apart, besides its many features, is that it automatically discovers the right configuration for your services.

Containers

[Kubernetes](#) is an open-source container-orchestration system for automating application deployment, scaling, and management. It was originally designed by Google, and is now maintained by the Cloud Native Computing Foundation.

[Google Kubernetes Engine \(GKE\)](#) is a managed, production-ready environment for deploying containerized applications.

[OpenShift](#) is focused on security at every level of the container stack and throughout the application lifecycle. It includes long-term, enterprise support from one of the leading Kubernetes contributors and open source software companies.

[Rancher](#) is a complete software stack for teams adopting containers. It addresses the operational and security challenges of managing multiple Kubernetes clusters, while providing DevOps teams with integrated tools for running containerized workloads.

[Docker](#) is a set of platform as a service products that use OS-level virtualization to deliver software in packages called containers. Containers are isolated from one another and bundle their own software, libraries and configuration files; they can communicate with each other through well-defined channels. All containers are run by a single operating-system kernel and are thus more lightweight than virtual machines.

[Rook](#) is an open source cloud-native storage orchestrator for Kubernetes that turns distributed storage systems into self-managing, self-scaling, self-healing storage services. It automates the tasks of a storage administrator: deployment, bootstrapping, configuration, provisioning, scaling, upgrading, migration, disaster recovery, monitoring, and resource management.

[Podman\(the POD MANager\)](#) is a tool for managing [OCI](#) containers and pods.

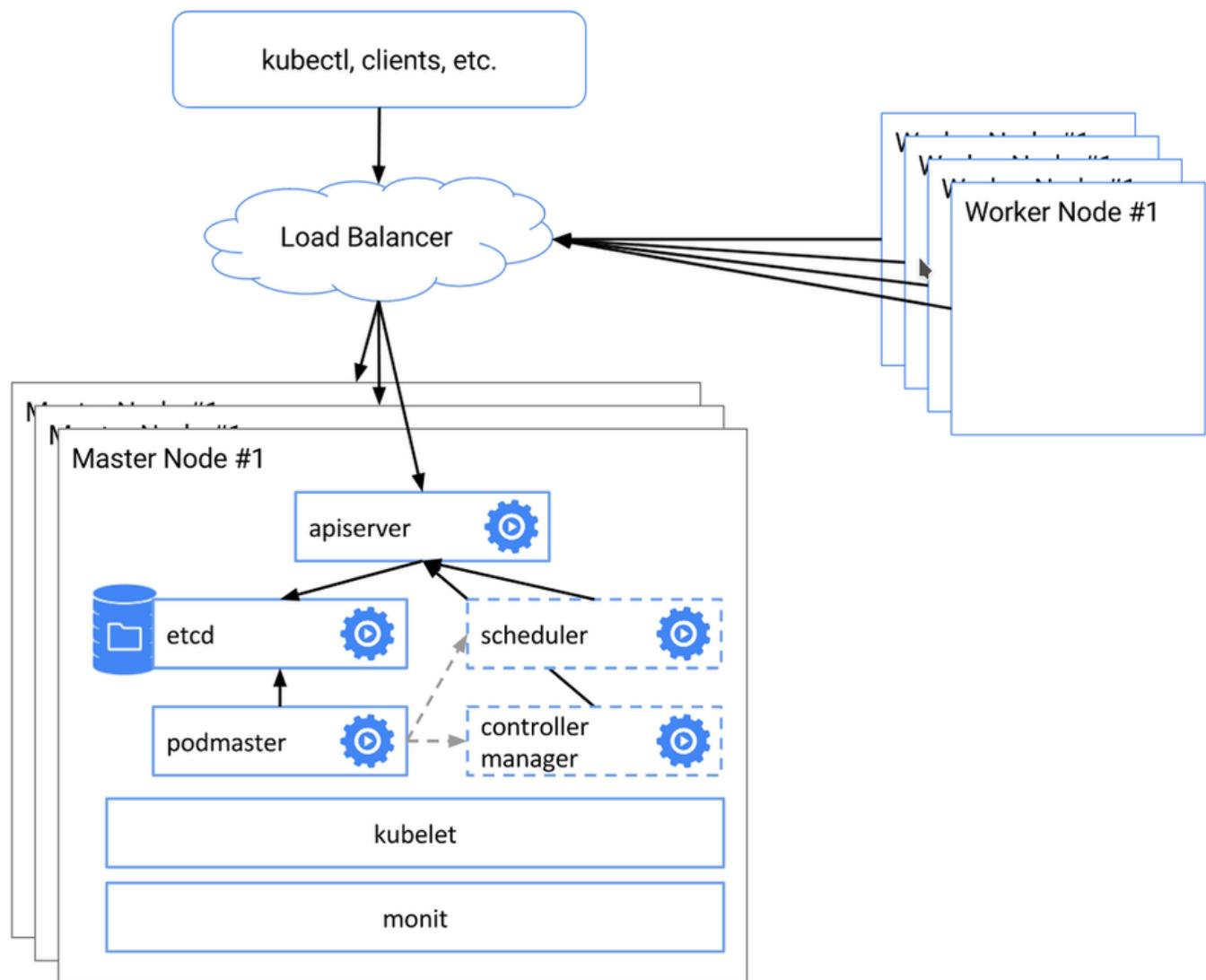
[Rkt](#) is a pod-native container engine for Linux. It is composable, secure, and built on standards.

7. Kubernetes

[Back to the Top](#)

Kubernetes Guide

A guide covering Kubernetes including the applications and tools that will make you a better and more efficient Kubernetes developer.



Building Highly-Availability(HA) Clusters with kubeadm. Source: [Kubernetes.io](https://kubernetes.io), 2020

Resources

Kubernetes (K8s) is an open-source system for automating deployment, scaling, and management of containerized applications.

[Getting Kubernetes Certifications](#)

[Getting started with Kubernetes on AWS](#)

[Kubernetes on Microsoft Azure](#)

[Intro to Azure Kubernetes Service](#)

[Azure Red Hat OpenShift](#)

[Getting started with Google Cloud](#)

[Getting started with Kubernetes on Red Hat](#)

[Getting started with Kubernetes on IBM](#)

[Red Hat OpenShift on IBM Cloud](#)

[Enable OpenShift Virtualization on Red Hat OpenShift](#)

[YAML basics in Kubernetes](#)

[Elastic Cloud on Kubernetes](#)

[Docker and Kubernetes](#)

[Running Apache Spark on Kubernetes](#)

[Kubernetes Across VMware vRealize Automation](#)

[VMware Tanzu Kubernetes Grid](#)

[All the Ways VMware Tanzu Works with AWS](#)

[VMware Tanzu Education](#)

[Using Ansible in a Cloud-Native Kubernetes Environment](#)

[Managing Kubernetes \(K8s\) objects with Ansible](#)

[Setting up a Kubernetes cluster using Vagrant and Ansible](#)

[Running MongoDB with Kubernetes](#)

[Kubernetes Fluentd](#)

[Understanding the new GitLab Kubernetes Agent](#)

[Intro Local Process with Kubernetes for Visual Studio 2019](#)

[Kubernetes Contributors](#)

[KubeAcademy from VMware](#)

[Kubernetes Tutorials from Pulumi](#)

[Kubernetes Playground by Katacoda](#)

Scalable Microservices with Kubernetes course from Udacity

Tools

[Google Kubernetes Engine \(GKE\)](#) is a managed, production-ready environment for running containerized applications.

[Azure Kubernetes Service \(AKS\)](#) is serverless Kubernetes, with a integrated continuous integration and continuous delivery (CI/CD) experience, and enterprise-grade security and governance. Unite your development and operations teams on a single platform to rapidly build, deliver, and scale applications with confidence.

[Amazon EKS](#) is a tool that runs Kubernetes control plane instances across multiple Availability Zones to ensure high availability.

[AWS Controllers for Kubernetes \(ACK\)](#) is a new tool that lets you directly manage AWS services from Kubernetes. ACK makes it simple to build scalable and highly-available Kubernetes applications that utilize AWS services.

[Container Engine for Kubernetes \(OKE\)](#) is an Oracle-managed container orchestration service that can reduce the time and cost to build modern cloud native applications. Unlike most other vendors, Oracle Cloud Infrastructure provides Container Engine for Kubernetes as a free service that runs on higher-performance, lower-cost compute.

[Anthos](#) is a modern application management platform that provides a consistent development and operations experience for cloud and on-premises environments.

[Red Hat OpenShift](#) is a fully managed Kubernetes platform that provides a foundation for on-premises, hybrid, and multicloud deployments.

[OKD](#) is a community distribution of Kubernetes optimized for continuous application development and multi-tenant deployment. OKD adds developer and operations-centric tools on top of Kubernetes to enable rapid application development, easy deployment and scaling, and long-term lifecycle maintenance for small and large teams.

[Odo](#) is a fast, iterative, and straightforward CLI tool for developers who write, build, and deploy applications on Kubernetes and OpenShift.

[Kata Operator](#) is an operator to perform lifecycle management (install/upgrade/uninstall) of [Kata Runtime](#) on OpenShift as well as Kubernetes cluster.

[Thanos](#) is a set of components that can be composed into a highly available metric system with unlimited storage capacity, which can be added seamlessly on top of existing Prometheus deployments.

[OpenShift Hive](#) is an operator which runs as a service on top of Kubernetes/OpenShift. The Hive service can be used to provision and perform initial configuration of OpenShift 4 clusters.

[Rook](#) is a tool that turns distributed storage systems into self-managing, self-scaling, self-healing storage services. It automates the tasks of a storage administrator: deployment, bootstrapping, configuration, provisioning, scaling, upgrading, migration, disaster recovery, monitoring, and resource management.

[VMware Tanzu](#) is a centralized management platform for consistently operating and securing your Kubernetes infrastructure and modern applications across multiple teams and private/public clouds.

[Kubespray](#) is a tool that combines Kubernetes and Ansible to easily install Kubernetes clusters that can be deployed on [AWS](#), [GCE](#), [Azure](#), [OpenStack](#), [vSphere](#), [Packet](#) (bare metal), Oracle Cloud Infrastructure (Experimental), or Baremetal.

[KubelInit](#) provides Ansible playbooks and roles for the deployment and configuration of multiple Kubernetes distributions.

[Rancher](#) is a complete software stack for teams adopting containers. It addresses the operational and security challenges of managing multiple Kubernetes clusters, while providing DevOps teams with integrated tools for running containerized workloads.

[K3s](#) is a highly available, certified Kubernetes distribution designed for production workloads in unattended, resource-constrained, remote locations or inside IoT appliances.

[Helm](#) is a Kubernetes Package Manager tool that makes it easier to install and manage Kubernetes applications.

[Knative](#) is a Kubernetes-based platform to build, deploy, and manage modern serverless workloads. Knative takes care of the operational overhead details of networking, autoscaling (even to zero), and revision tracking.

[KubeFlow](#) is a tool dedicated to making deployments of machine learning (ML) workflows on Kubernetes simple, portable and scalable.

[Etcd](#) is a distributed key-value store that provides a reliable way to store data that needs to be accessed by a distributed system or cluster of machines. Etcd is used as the backend for service discovery and stores cluster state and configuration for Kubernetes.

[OpenEBS](#) is a Kubernetes-based tool to create stateful applications using Container Attached Storage.

[Container Storage Interface \(CSI\)](#) is an API that lets container orchestration platforms like Kubernetes seamlessly communicate with stored data via a plug-in.

[MicroK8s](#) is a tool that delivers the full Kubernetes experience. In a Fully containerized deployment with compressed over-the-air updates for ultra-reliable operations. It is supported on Linux, Windows, and MacOS.

[Charmed Kubernetes](#) is a well integrated, turn-key, conformant Kubernetes platform, optimized for your multi-cloud environments developed by Canonical.

[Grafana Kubernetes App](#) is a toll that allows you to monitor your Kubernetes cluster's performance. It includes 4 dashboards, Cluster, Node, Pod/Container and Deployment. It allows for the automatic deployment of the required Prometheus exporters and a default scrape config to use with your in cluster Prometheus deployment.

[KubeEdge](#) is an open source system for extending native containerized application orchestration capabilities to hosts at Edge. It is built upon kubernetes and provides fundamental infrastructure support for network, app. deployment and metadata synchronization between cloud and edge.

[Lens](#) is the most powerful IDE for people who need to deal with Kubernetes clusters on a daily basis. It has support for MacOS, Windows and Linux operating systems.

[Kind](#) is a tool for running local Kubernetes clusters using Docker container "nodes". It was primarily designed for testing Kubernetes itself, but may be used for local development or CI.

[Flux CD](#) is a tool that automatically ensures that the state of your Kubernetes cluster matches the configuration you've supplied in Git. It uses an operator in the cluster to trigger deployments inside Kubernetes, which means that you don't need a separate continuous delivery tool.

[Platform9 Managed Kubernetes \(PMK\)](#) is a Kubernetes as a service that ensures fully automated Day-2 operations with 99.9% SLA on any environment, whether in data-centers, public clouds, or at the edge.

8. ARM

[Back to the Top](#)



ARM Guide

A guide covering ARM architecture including the applications and tools that will make you a better and more efficient ARM developer.

ARM Resources

[ARM](#) stands for Advanced RISC Machine, which is a collection of reduced instruction set computing architectures for CPUs configured for various development environments such servers, IoT, and other mobile devices. Arm was acquired by [NVIDIA](#) in [September 2020](#).

[NVIDIA CUDA on Arm](#)

[Learning the Arm Architecture](#)

[Arm Development Tools and Software](#)

[GNU Arm Embedded Toolchain](#)

[Internet of Things on Arm](#)

[MATLAB Code Generation for Deep Learning on ARM Targets](#)

[Getting started with AWS Graviton](#)

[Improving performance of PHP for Arm64 and impact on AWS Graviton2 based EC2 instances](#)

[Google's "Whitechapel" 5nm SoC Chip for future Pixel devices & Chromebooks](#)

Qualcomm Snapdragon Mobile Platforms, Processors and Chipsets

[Windows 10 on ARM](#)

[Introducing x64 emulation in preview for Windows 10 on ARM PCs to the Windows Insider Program](#)

[Porting UWP applications for Windows 10 on ARM](#)

[Configuring C++ projects for ARM processors using the Microsoft Visual C++ \(MSVC\) compiler toolset](#)

[Run ARM applications on the Android Emulator](#)

[Android Application Binary Interface \(ABIs\)](#)

[Ubuntu Server for ARM](#)

[Fedora ARM](#)

[Manjaro Linux ARM](#)

[SUSE Linux Enterprise Server for Arm](#)

[openSUSE Arm](#)

[Raspberry Pi board computers](#)

[NVIDIA Jetson Nano 2GB Developer Kit](#)

[Arm Online Courses](#)

[Deep Learning on ARM Processors - From Ground Up online course](#)

[Embedded Systems using the ARM Mbed Platform online course](#)

[Microcontroller Courses from Coursera](#)

Apple Silicon Resources



[Does it ARM? Apps that are reported to support Apple Silicon](#)

[Apple Hypervisor](#) is a framework that builds virtualization solutions on top of a lightweight hypervisor, without third-party kernel extensions. Hypervisor provides C APIs so you can interact with virtualization technologies in user space, without writing kernel extensions (KEXTs). As a result, the apps you create using this framework are suitable for distribution on the [Mac App Store](#).

[Apple A-series](#) is Apple's 64-bit ARM-based system on a chip (SoC) used in their iPhones and iPads. Though, at WWDC 2020 it was announced that [Apple Silicon](#) would [transition into Mac laptops](#).

[Apple M1 Chip](#) is Apple's first SoC chip designed specifically for their ARM Mac products, it delivers incredible performance(8-core CPU and 8-core GPU), custom technologies, and great power efficiency. The M1 Chip is now available for [Macbook Pro 13 with M1](#), [Macbook Air 13 with M1](#), and [Mac Mini with M1](#).

[Xcode 12](#) is built as an Universal app that runs 100% natively on Intel-based CPUs and Apple Silicon. It includes a unified macOS SDK that features all the frameworks, compilers, debuggers, and other tools you need to build apps that run natively on Apple Silicon and the Intel x86_64 CPU.

[Tensorflow_macOS](#) is a Mac-optimized version of TensorFlow and TensorFlow Addons for macOS 11.0+ accelerated using Apple's ML Compute framework.

[Universal App Quick Start Program](#)

[Writing ARM64 Code for Apple Platforms](#)

[Porting Your macOS Apps to Apple Silicon](#)

[Building a Universal macOS Binary](#)

[Addressing Architectural Differences in Your macOS Code](#)

[Porting Just-In-Time\(JIT\) Compilers to Apple Silicon](#)

[Porting Your Audio Code to Apple Silicon](#)

[Porting Your Metal Code to Apple Silicon](#)

[Tuning Your Code's Performance for Apple Silicon](#)

[Learn how Rosetta translates executables and what Rosetta can't translate](#)

[Running Your iOS Apps on macOS](#)

[Adapting iOS Code to Run in the macOS Environment](#)

[Implementing Drivers, System Extensions, and Kexts](#)

[Installing a Custom Kernel Extension](#)

[Debugging a Custom Kernel Extension](#)

Tools

[Arm Instruction Emulator \(ArmIE\)](#) is a tool that emulates Scalable Vector Extension (SVE) and SVE2 instructions on AArch64/ARM64 platforms.

[Arm Mobile Studio](#) is a suite of free-to-use performance analysis tools that automatically analyzes the CPU activity, GPU activity and content metrics of your game as it runs on a non-rooted Android device.

[Android Studio](#) is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Available on Windows, macOS, Linux, Chrome OS.

[Visual Studio](#) is an integrated development environment (IDE) from Microsoft; which is a feature-rich application that can be used for many aspects of software development. Visual Studio makes it easy to edit, debug, build, and publish your app. By using Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, and Windows Store.

[Arduino IDE](#) is an open-source integrated development environment for the Arduino platform that provides easy-to-use hardware and software.

[Compute Library](#) is a set of computer vision and machine learning functions optimised for both Arm CPUs and GPUs using SIMD technologies.

[LISA](#) is a project provides a toolkit that supports regression testing and interactive analysis of Linux kernel behavior. LISA stands for Linux Integrated/Interactive System Analysis. LISA's goal is to help Linux kernel developers to measure the impact of modifications in core parts of the kernel. The focus is on the scheduler (EAS), power management and thermal frameworks.

[LLVM](#) is a library that has a collection of modular/reusable compiler and toolchain components(assemblers, compilers, debuggers, etc.). With these components LLVM can be used as a compiler framework, providing a front-end(parser and lexer) and a back-end(where code converts LLVM's representation to actual machine code).

[The Eclipse Embedded CDT](#) is a collection of plug-ins for Arm & RISC-V C/C++ developers.

[PlatformIO](#) is a professional collaborative platform for embedded development with no vendor lock-in. It provides support for multiplatforms and frameworks such as IoT, Arduino, CMSIS, ESP-IDF, FreeRTOS, libOpenCM3, mbed OS, Pulp OS, SPL, STM32Cube, Zephyr RTOS, ARM, AVR, Espressif (ESP8266/ESP32), FPGA, MCS-51 (8051), MSP430, Nordic (nRF51/nRF52), NXP i.MX RT, PIC32, RISC-V.

[PlatformIO for VSCode](#) is a plugin that provides support for the PlatformIO IDE on VSCode.

[simdjson](#) library uses commonly available SIMD instructions and microparallel algorithms to parse gigabytes of JSON per second.

[TinyGo](#) is a Go compiler(based on LLVM) intended for use in small places such as microcontrollers, WebAssembly (Wasm), and command-line tools.

[Unicorn](#) is a lightweight, multi-platform, multi-architecture CPU emulator framework(ARM, AArch64, M68K, Mips, Sparc, X86) based on [QEMU](#).

[Tock](#) is an embedded operating system designed for running multiple concurrent, mutually distrustful applications on Cortex-M and RISC-V based embedded platforms. Tock's design centers around protection, both from potentially malicious applications and from device drivers.

[Keystone](#) is a lightweight multi-platform, multi-architecture(Arm, Arm64, Hexagon, Mips, PowerPC, Sparc, SystemZ & X86) assembler framework.

[Faasd](#) is a project similar to [OpenFaaS](#), but without the cost and complexity of Kubernetes. It runs on a single host with very modest requirements, making it fast and easy to manage. Under the hood it uses [containerd](#) and [Container Networking Interface \(CNI\)](#) along with the same core OpenFaaS components from the main project.

9. Assembly(x86)

[Back to the Top](#)



Assembly Guide

A guide covering Assembly programming language including the x86 computer architectures, applications and tools that will make you a better and more efficient developer.

Resources

[Assembly](#) is a low-level programming language. It uses mnemonic codes and labels to represent machine-level code with each instruction corresponding to just one machine operation.

[RISC-V Foundation](#) is a non-profit corporation controlled by its 500 members(NVIDIA, Google, Samsung, Raspberry Pi, SiFive, Canonical, and Western Digital) to drive forward the adoption and implementation of the free and open RISC-V instruction set architecture (ISA).

[Intel® 64 and IA-32 Architectures Software Developer's Manual](#)

[Introduction to x64 Assembly from Intel](#)

[x86 Assembly Language Reference Manual for Open Solaris](#)

[AMD64 Architecture Programmer's Manual Volume 1-5](#)

[AMD GPU ISA documentation](#)

[AMD Developer Guides, Manuals, and ISA Documents](#)

[Assembler language from IBM](#)

[The Assembler language on z/OS from IBM](#)

[MIPS Architecture & Technology from Wave Computing](#)

[Assemblies in .NET](#)

[Microsoft Macro Assembler reference](#)

[Compiler Intrinsics and Assembly Language from Microsoft](#)

[x86 and amd64 instruction Reference](#)

[Intro to x86 Assembly Language Programming](#)

[Learn Assembly Programming courses on Udemy](#)

[Assembly Languages and Assemblers courses on Coursera](#)

[Intro to Assembly Language from MIT](#)

Tools

[Arm Instruction Emulator \(ArmIE\)](#) is a tool that emulates Scalable Vector Extension (SVE) and SVE2 instructions on AArch64/ARM64 platforms.

[FASM \(flat assembler\)](#) is an assembler for x86 processors that supports Intel-based assembly language on the IA-32 and x86-64 computer architectures.

[Microsoft Assembler \(MASM\) for x64](#) is Microsoft's assembler that accepts x64 assembler language.

[MASM/TASM](#) is a VSCode extension that offers a way to run and debug DOS(80x86) assembly TASM/MASM through DOSBox and msdos-player.

[NASM](#) is an assembler/disassembler for the x86 CPU architecture portable to nearly every modern platform, and with code generation for many platforms old and new.

[GAS](#) is the assembler used by the GNU Project for the default back-end of GCC. It is used to assemble the GNU operating system and the Linux kernel.

[MIPS](#) is a reduced instruction set computer (RISC) instruction set architecture (ISA) developed by [MIPS Technologies, Inc.](#). In June 2018 [MIPS was Acquired by AI Startup Wave Computing](#).

[LLVM](#) is a library that has collection of modular/reusable compiler and toolchain components (assemblers, compilers, debuggers, etc.). With these components LLVM can be used as a compiler framework, providing a front-end(parser and lexer) and a back-end (code that converts LLVM's representation to actual machine code).

[TinyGo](#) is a Go compiler(based on LLVM) intended for use in small places such as microcontrollers, WebAssembly (Wasm), and command-line tools.

[Tock](#) is an embedded operating system designed for running multiple concurrent, mutually distrustful applications on Cortex-M and RISC-V based embedded platforms. Tock's design centers around protection, both from potentially malicious applications and from device drivers.

[PlatformIO](#) is a professional collaborative platform for embedded development with no vendor lock-in. It provides support for multiplatforms and frameworks such as IoT, Arduino, CMSIS, ESP-IDF, FreeRTOS, libOpenCM3, mbed OS, Pulp OS, SPL, STM32Cube, Zephyr RTOS, ARM, AVR, Espressif (ESP8266/ESP32), FPGA, MCS-51 (8051), MSP430, Nordic (nRF51/nRF52), NXP i.MX RT, PIC32, RISC-V.

[PlatformIO for VSCode](#) is a plugin that provides support for the PlatformIO IDE on VSCode.

[Keystone](#) is a lightweight multi-platform, multi-architecture(Arm, Arm64, Hexagon, Mips, PowerPC, Sparc, SystemZ & X86) assembler framework.

[Unicorn](#) is a lightweight, multi-platform, multi-architecture CPU emulator framework(ARM, AArch64, M68K, Mips, Sparc, X86) based on [QEMU](#).

10. Machine Learning

[Back to the Top](#)



Machine Learning Guide

A guide covering Machine Learning including the applications, libraries and tools that will make you better and more efficient with Machine Learning development.



Machine Learning/Deep Learning Frameworks.

Resources

Machine Learning is a branch of artificial intelligence (AI) focused on building apps using algorithms that learn from data models and improve their accuracy over time without needing to be programmed.

[Machine Learning by Stanford University from Coursera](#)

[AWS Training and Certification for Machine Learning \(ML\) Courses](#)

[Machine Learning Scholarship Program for Microsoft Azure from Udacity](#)

[Microsoft Certified: Azure Data Scientist Associate](#)

[Microsoft Certified: Azure AI Engineer Associate](#)

[Azure Machine Learning training and deployment](#)

[Learning Machine learning and artificial intelligence from Google Cloud Training](#)

[Machine Learning Crash Course for Google Cloud](#)

[JupyterLab](#)

[Scheduling Jupyter notebooks on Amazon SageMaker ephemeral instances](#)

[How to run Jupyter Notebooks in your Azure Machine Learning workspace](#)

[Machine Learning Courses Online from Udemy](#)

[Machine Learning Courses Online from Coursera](#)

[Learn Machine Learning with Online Courses and Classes from edX](#)

Tools

[TensorFlow](#) is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

[Keras](#) is a high-level neural networks API, written in Python and capable of running on top of TensorFlow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, R, Theano, or PlaidML.

[PyTorch](#) is a library for deep learning on irregular input data such as graphs, point clouds, and manifolds. Primarily developed by Facebook's AI Research lab.

[Amazon SageMaker](#) is a fully managed service that provides every developer and data scientist with the ability to build, train, and deploy machine learning (ML) models quickly. SageMaker removes the heavy lifting from each step of the machine learning process to make it easier to develop high quality models.

[Azure Databricks](#) is a fast and collaborative Apache Spark-based big data analytics service designed for data science and data engineering. Azure Databricks, sets up your Apache Spark environment in minutes, autoscale, and collaborate on shared projects in an interactive workspace. Azure Databricks supports Python, Scala, R, Java, and SQL, as well as data science frameworks and libraries including TensorFlow, PyTorch, and scikit-learn.

[Microsoft Cognitive Toolkit \(CNTK\)](#) is an open-source toolkit for commercial-grade distributed deep learning. It describes neural networks as a series of computational steps via a directed graph. CNTK allows the user to easily realize and combine popular model types such as feed-forward DNNs, convolutional neural networks (CNNs) and recurrent neural networks (RNNs/LSTMs). CNTK implements stochastic gradient descent (SGD, error backpropagation) learning with automatic differentiation and parallelization across multiple GPUs and servers.

[Apple CoreML](#) is a framework that helps integrate machine learning models into your app. Core ML provides a unified representation for all models. Your app uses Core ML APIs and user data to make

predictions, and to train or fine-tune models, all on the user's device. A model is the result of applying a machine learning algorithm to a set of training data. You use a model to make predictions based on new input data.

[Tensorflow_macOS](#) is a Mac-optimized version of TensorFlow and TensorFlow Addons for macOS 11.0+ accelerated using Apple's ML Compute framework.

[Apache Airflow](#) is an open-source workflow management platform created by the community to programmatically author, schedule and monitor workflows. Install. Principles. Scalable. Airflow has a modular architecture and uses a message queue to orchestrate an arbitrary number of workers. Airflow is ready to scale to infinity.

[Open Neural Network Exchange\(ONNX\)](#) is an open ecosystem that empowers AI developers to choose the right tools as their project evolves. ONNX provides an open source format for AI models, both deep learning and traditional ML. It defines an extensible computation graph model, as well as definitions of built-in operators and standard data types.

[Apache MXNet](#) is a deep learning framework designed for both efficiency and flexibility. It allows you to mix symbolic and imperative programming to maximize efficiency and productivity. At its core, MXNet contains a dynamic dependency scheduler that automatically parallelizes both symbolic and imperative operations on the fly. A graph optimization layer on top of that makes symbolic execution fast and memory efficient. MXNet is portable and lightweight, scaling effectively to multiple GPUs and multiple machines. Support for Python, R, Julia, Scala, Go, Javascript and more.

[AutoGluon](#) is toolkit for Deep learning that automates machine learning tasks enabling you to easily achieve strong predictive performance in your applications. With just a few lines of code, you can train and deploy high-accuracy deep learning models on tabular, image, and text data.

[Anaconda](#) is a very popular Data Science platform for machine learning and deep learning that enables users to develop models, train them, and deploy them.

[PlaidML](#) is an advanced and portable tensor compiler for enabling deep learning on laptops, embedded devices, or other devices where the available computing hardware is not well supported or the available software stack contains unpalatable license restrictions.

[OpenCV](#) is a highly optimized library with focus on real-time computer vision applications. The C++, Python, and Java interfaces support Linux, MacOS, Windows, iOS, and Android.

[Scikit-Learn](#) is a Python module for machine learning built on top of SciPy, NumPy, and matplotlib, making it easier to apply robust and simple implementations of many popular machine learning algorithms.

[Weka](#) is an open source machine learning software that can be accessed through a graphical user interface, standard terminal applications, or a Java API. It is widely used for teaching, research, and industrial applications, contains a plethora of built-in tools for standard machine learning tasks, and additionally gives transparent access to well-known toolboxes such as scikit-learn, R, and Deeplearning4j.

[Caffe](#) is a deep learning framework made with expression, speed, and modularity in mind. It is developed by Berkeley AI Research (BAIR)/The Berkeley Vision and Learning Center (BVLC) and community contributors.

[Theano](#) is a Python library that allows you to define, optimize, and evaluate mathematical expressions involving multi-dimensional arrays efficiently including tight integration with NumPy.

[nGraph](#) is an open source C++ library, compiler and runtime for Deep Learning. The nGraph Compiler aims to accelerate developing AI workloads using any deep learning framework and deploying to a variety of hardware targets. It provides the freedom, performance, and ease-of-use to AI developers.

[NVIDIA cuDNN](#) is a GPU-accelerated library of primitives for [deep neural networks](#). cuDNN provides highly tuned implementations for standard routines such as forward and backward convolution, pooling, normalization, and activation layers. cuDNN accelerates widely used deep learning frameworks, including [Caffe2](#), [Chainer](#), [Keras](#), [MATLAB](#), [MxNet](#), [PyTorch](#), and [TensorFlow](#).

[Jupyter Notebook](#) is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Jupyter is used widely in industries that do data cleaning and transformation, numerical simulation, statistical modeling, data visualization, data science, and machine learning.

[Apache Spark](#) is a unified analytics engine for large-scale data processing. It provides high-level APIs in Scala, Java, Python, and R, and an optimized engine that supports general computation graphs for data analysis. It also supports a rich set of higher-level tools including Spark SQL for SQL and DataFrames, MLlib for machine learning, GraphX for graph processing, and Structured Streaming for stream processing.

[Apache Spark Connector for SQL Server and Azure SQL](#) is a high-performance connector that enables you to use transactional data in big data analytics and persists results for ad-hoc queries or reporting. The connector allows you to use any SQL database, on-premises or in the cloud, as an input data source or output data sink for Spark jobs.

[Apache PredictionIO](#) is an open source machine learning framework for developers, data scientists, and end users. It supports event collection, deployment of algorithms, evaluation, querying predictive results via REST APIs. It is based on scalable open source services like Hadoop, HBase (and other DBs), Elasticsearch, Spark and implements what is called a Lambda Architecture.

[Cluster Manager for Apache Kafka\(CMAK\)](#) is a tool for managing [Apache Kafka](#) clusters.

[BigDL](#) is a distributed deep learning library for Apache Spark. With BigDL, users can write their deep learning applications as standard Spark programs, which can directly run on top of existing Spark or Hadoop clusters.

[Eclipse Deeplearning4J \(DL4J\)](#) is a set of projects intended to support all the needs of a JVM-based(Scala, Kotlin, Clojure, and Groovy) deep learning application. This means starting with the raw data, loading and preprocessing it from wherever and whatever format it is in to building and tuning a wide variety of simple and complex deep learning networks.

[Tensorman](#) is a utility for easy management of Tensorflow containers by developed by [System76](#). Tensorman allows Tensorflow to operate in an isolated environment that is contained from the rest of the system. This virtual environment can operate independent of the base system, allowing you to use any version of Tensorflow on any version of a Linux distribution that supports the Docker runtime.

[Numba](#) is an open source, NumPy-aware optimizing compiler for Python sponsored by Anaconda, Inc. It uses the LLVM compiler project to generate machine code from Python syntax. Numba can compile a large subset of numerically-focused Python, including many NumPy functions. Additionally, Numba has support for automatic parallelization of loops, generation of GPU-accelerated code, and creation of ufuncs and C callbacks.

[Chainer](#) is a Python-based deep learning framework aiming at flexibility. It provides automatic differentiation APIs based on the define-by-run approach (dynamic computational graphs) as well as object-oriented high-level APIs to build and train neural networks. It also supports CUDA/cuDNN using [CuPy](#) for high performance training and inference.

[cuML](#) is a suite of libraries that implement machine learning algorithms and mathematical primitives functions that share compatible APIs with other RAPIDS projects. cuML enables data scientists, researchers, and software engineers to run traditional tabular ML tasks on GPUs without going into the details of CUDA programming. In most cases, cuML's Python API matches the API from scikit-learn.

11. Python Guide

[Back to the Top](#)



Python Guide

A guide covering the Python programming language including the applications and tools that will make you a better and more efficient Python developer.

Resources

[Python](#) is an interpreted, high-level programming language. Python is used heavily in the fields of Data Science and Machine Learning.

[Python Developer's Guide](#) is a comprehensive resource for contributing to Python – for both new and experienced contributors. It is maintained by the same community that maintains Python.

[Azure Functions Python developer guide](#) is an introduction to developing Azure Functions using Python. The content below assumes that you've already read the [Azure Functions developers guide](#).

[CheckiO](#) is a programming learning platform and a gamified website that teaches Python through solving code challenges and competing for the most elegant and creative solutions.

[Python Institute](#)

[PCEP – Certified Entry-Level Python Programmer certification](#)

[PCAP – Certified Associate in Python Programming certification](#)

[PCPP – Certified Professional in Python Programming 1 certification](#)

[PCPP – Certified Professional in Python Programming 2](#)

[MTA: Introduction to Programming Using Python Certification](#)

[Getting Started with Python in Visual Studio Code](#)

[Google's Python Style Guide](#)

[Google's Python Education Class](#)

[Real Python](#)

[The Python Open Source Computer Science Degree by Forrest Knight](#)

[Intro to Python for Data Science](#)

[Intro to Python by W3schools](#)

[Codecademy's Python 3 course](#)

[Learn Python with Online Courses and Classes from edX](#)

[Python Courses Online from Coursera](#)

Tools

[Python Package Index \(PyPI\)](#) is a repository of software for the Python programming language. PyPI helps you find and install software developed and shared by the Python community.

[PyCharm](#) is the best IDE I've ever used. With PyCharm, you can access the command line, connect to a database, create a virtual environment, and manage your version control system all in one place, saving time by avoiding constantly switching between windows.

[Python Tools for Visual Studio\(PTVS\)](#) is a free, open source plugin that turns Visual Studio into a Python IDE. It supports editing, browsing, IntelliSense, mixed Python/C++ debugging, remote Linux/MacOS debugging, profiling, IPython, and web development with Django and other frameworks.

[Pylance](#) is an extension that works alongside Python in Visual Studio Code to provide performant language support. Under the hood, Pylance is powered by Pyright, Microsoft's static type checking tool.

[Pyright](#) is a fast type checker meant for large Python source bases. It can run in a "watch" mode and performs fast incremental updates when files are modified.

[Django](#) is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.

[Flask](#) is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries.

[Web2py](#) is an open-source web application framework written in Python allowing allows web developers to program dynamic web content. One web2py instance can run multiple web sites using different databases.

[AWS Chalice](#) is a framework for writing serverless apps in python. It allows you to quickly create and deploy applications that use AWS Lambda.

[Tornado](#) is a Python web framework and asynchronous networking library. Tornado uses a non-blocking network I/O, which can scale to tens of thousands of open connections.

[HTTPie](#) is a command line HTTP client that makes CLI interaction with web services as easy as possible. HTTPie is designed for testing, debugging, and generally interacting with APIs & HTTP servers.

[Scrapy](#) is a fast high-level web crawling and web scraping framework, used to crawl websites and extract structured data from their pages. It can be used for a wide range of purposes, from data mining to monitoring and automated testing.

[Sentry](#) is a service that helps you monitor and fix crashes in realtime. The server is in Python, but it contains a full API for sending events from any language, in any application.

[Pipenv](#) is a tool that aims to bring the best of all packaging worlds (bundler, composer, npm, cargo, yarn, etc.) to the Python world.

[Python Fire](#) is a library for automatically generating command line interfaces (CLIs) from absolutely any Python object.

[Bottle](#) is a fast, simple and lightweight [WSGI](#) micro web-framework for Python. It is distributed as a single file module and has no dependencies other than the [Python Standard Library](#).

[CherryPy](#) is a minimalist Python object-oriented HTTP web framework.

[Sanic](#) is a Python 3.6+ web server and web framework that's written to go fast.

[Pyramid](#) is a small and fast open source Python web framework. It makes real-world web application development and deployment more fun and more productive.

[TurboGears](#) is a hybrid web framework able to act both as a Full Stack framework or as a Microframework.

[Falcon](#) is a reliable, high-performance Python web framework for building large-scale app backends and microservices with support for MongoDB, Pluggable Applications and autogenerated Admin.

[Neural Network Intelligence\(NNI\)](#) is an open source AutoML toolkit for automate machine learning lifecycle, including [Feature Engineering](#), [Neural Architecture Search](#), [Model Compression](#) and [Hyperparameter Tuning](#).

[Dash](#) is a popular Python framework for building ML & data science web apps for Python, R, Julia, and Jupyter.

[Luigi](#) is a Python module that helps you build complex pipelines of batch jobs. It handles dependency resolution, workflow management, visualization etc. It also comes with Hadoop support built-in.

[Locust](#) is an easy to use, scriptable and scalable performance testing tool.

[spaCy](#) is a library for advanced Natural Language Processing in Python and Cython.

[NumPy](#) is the fundamental package needed for scientific computing with Python.

[Pillow](#) is a friendly PIL(Python Imaging Library) fork.

[IPython](#) is a command shell for interactive computing in multiple programming languages, originally developed for the Python programming language, that offers enhanced introspection, rich media, additional shell syntax, tab completion, and rich history.

[GraphLab Create](#) is a Python library, backed by a C++ engine, for quickly building large-scale, high-performance machine learning models.

[Pandas](#) is a fast, powerful, and easy to use open source data structures, data analysis and manipulation tool, built on top of the Python programming language.

[PuLP](#) is an Linear Programming modeler written in python. PuLP can generate LP files and call on use highly optimized solvers, GLPK, COIN CLP/CBC, CPLEX, and GUROBI, to solve these linear problems.

[Matplotlib](#) is a 2D plotting library for creating static, animated, and interactive visualizations in Python. Matplotlib produces publication-quality figures in a variety of hardcopy formats and interactive environments across platforms.

[Scikit-Learn](#) is a simple and efficient tool for data mining and data analysis. It is built on NumPy,SciPy, and matplotlib.

12. SQL

[Back to the Top](#)



SQL Guide

A guide covering the SQL programming language including the applications and tools that will make you a better and more efficient SQL developer.

Resources

[SQL](#) is a standard language for storing, manipulating and retrieving data in relational databases.

[SQL Tutorial by W3Schools](#)

[Learn SQL Skills Online from Coursera](#)

[SQL Courses Online from Udemy](#)

[SQL Online Training Courses from LinkedIn Learning](#)

[Learn SQL For Free from Codecademy](#)

[GitLab's SQL Style Guide](#)

[OracleDB SQL Style Guide Basics](#)

[Tableau CRM: BI Software and Tools](#)

Databases on AWS

[Best Practices and Recommendations for SQL Server Clustering in AWS EC2.](#)

[Connecting from Google Kubernetes Engine to a Cloud SQL instance.](#)

[Educational Microsoft Azure SQL resources](#)

[MySQL Certifications](#)

[SQL vs. NoSQL Databases: What's the Difference?](#)

[What is NoSQL?](#)

Tools

[Azure Data Studio](#) is an open source data management tool that enables working with SQL Server, Azure SQL DB and SQL DW from Windows, macOS and Linux.

[Azure SQL Database](#) is the intelligent, scalable, relational database service built for the cloud. It's evergreen and always up to date, with AI-powered and automated features that optimize performance and durability for you. Serverless compute and Hyperscale storage options automatically scale resources on demand, so you can focus on building new applications without worrying about storage size or resource management.

[Azure SQL Managed Instance](#) is a fully managed SQL Server Database engine instance that's hosted in Azure and placed in your network. This deployment model makes it easy to lift and shift your on-premises applications to the cloud with very few application and database changes. Managed instance has split compute and storage components.

[Azure Synapse Analytics](#) is a limitless analytics service that brings together enterprise data warehousing and Big Data analytics. It gives you the freedom to query data on your terms, using either serverless or provisioned resources at scale. It brings together the best of the SQL technologies used in enterprise data warehousing, Spark technologies used in big data analytics, and Pipelines for data integration and ETL/ELT.

[MSSQL for Visual Studio Code](#) is an extension for developing Microsoft SQL Server, Azure SQL Database and SQL Data Warehouse everywhere with a rich set of functionalities.

[SQL Server Data Tools \(SSDT\)](#) is a development tool for building SQL Server relational databases, Azure SQL Databases, Analysis Services (AS) data models, Integration Services (IS) packages, and Reporting Services (RS) reports. With SSDT, a developer can design and deploy any SQL Server content type with the same ease as they would develop an application in Visual Studio or Visual Studio Code.

[Bulk Copy Program](#) is a command-line tool that comes with Microsoft SQL Server. BCP, allows you to import and export large amounts of data in and out of SQL Server databases quickly and efficiently.

[SQL Server Migration Assistant](#) is a tool from Microsoft that simplifies database migration process from Oracle to SQL Server, Azure SQL Database, Azure SQL Database Managed Instance and Azure SQL Data Warehouse.

[SQL Server Integration Services](#) is a development platform for building enterprise-level data integration and data transformations solutions. Use Integration Services to solve complex business problems by

copying or downloading files, loading data warehouses, cleansing and mining data, and managing SQL Server objects and data.

[SQL Server Business Intelligence\(BI\)](#) is a collection of tools in Microsoft's SQL Server for transforming raw data into information businesses can use to make decisions.

[Tableau](#) is a Data Visualization software used in relational databases, cloud databases, and spreadsheets. Tableau was acquired by [Salesforce in August 2019](#).

[DataGrip](#) is a professional DataBase IDE developed by Jet Brains that provides context-sensitive code completion, helping you to write SQL code faster. Completion is aware of the tables structure, foreign keys, and even database objects created in code you're editing.

[RStudio](#) is an integrated development environment for R and Python, with a console, syntax-highlighting editor that supports direct code execution, and tools for plotting, history, debugging and workspace management.

[MySQL](#) is a fully managed database service to deploy cloud-native applications using the world's most popular open source database.

[PostgreSQL](#) is a powerful, open source object-relational database system with over 30 years of active development that has earned it a strong reputation for reliability, feature robustness, and performance.

[Amazon DynamoDB](#) is a key-value and document database that delivers single-digit millisecond performance at any scale. It is a fully managed, multiregion, multimaster, durable database with built-in security, backup and restore, and in-memory caching for internet-scale applications.

[FoundationDB](#) is an open source distributed database designed to handle large volumes of structured data across clusters of commodity servers. It organizes data as an ordered key-value store and employs ACID transactions for all operations. It is especially well-suited for read/write workloads but also has excellent performance for write-intensive workloads. FoundationDB was acquired by [Apple in 2015](#).

[CouchbaseDB](#) is an open source distributed [multi-model NoSQL document-oriented database](#). It creates a key-value store with managed cache for sub-millisecond data operations, with purpose-built indexers for efficient queries and a powerful query engine for executing SQL queries.

[IBM DB2](#) is a collection of hybrid data management products offering a complete suite of AI-empowered capabilities designed to help you manage both structured and unstructured data on premises as well as in private and public cloud environments. Db2 is built on an intelligent common SQL engine designed for scalability and flexibility.

[MongoDB](#) is a document database meaning it stores data in JSON-like documents.

[OracleDB](#) is a powerful fully managed database helps developers manage business-critical data with the highest availability, reliability, and security.

[MariaDB](#) is an enterprise open source database solution for modern, mission-critical applications.

[SQLite](#) is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine.SQLite is the most used database engine in the world. SQLite is built into all mobile

phones and most computers and comes bundled inside countless other applications that people use every day.

[SQLite Database Browser](#) is an open source SQL tool that allows users to create, design and edits SQLite database files. It lets users show a log of all the SQL commands that have been issued by them and by the application itself.

[SolarWinds Database Performance Analyzer](#) is a performance monitoring and analysis tool for database admin. It also finds the cause of bottlenecks and reduces the overall cost of database operations.

[dbWatch](#) is a complete database monitoring/management solution for SQL Server, Oracle, PostgreSQL, Sybase, MySQL and Azure. Designed for proactive management and automation of routine maintenance in large scale on-premise, hybrid/cloud database environments.

[Cosmos DB Profiler](#) is a real-time visual debugger allowing a development team to gain valuable insight and perspective into their usage of Cosmos DB database. It identifies over a dozen suspicious behaviors from your application's interaction with Cosmos DB.

[Adminer](#) is an SQL management client tool for managing databases, tables, relations, indexes, users. Adminer has support for all the popular database management systems such as MySQL, MariaDB, PostgreSQL, SQLite, MS SQL, Oracle, Firebird, SimpleDB, Elasticsearch and MongoDB.

[DBeaver](#) is an open source database tool for developers and database administrators. It offers supports for JDBC compliant databases such as MySQL, Oracle, IBM DB2, SQL Server, Firebird, SQLite, Sybase, Teradata, Firebird, Apache Hive, Phoenix, and Presto.

[DbVisualizer](#) is a SQL management tool that allows users to manage a wide range of databases such as Oracle, Sybase, SQL Server, MySQL, H3, and SQLite.

[AppDynamics Database](#) is a management product for Microsoft SQL Server. With AppDynamics you can monitor and trend key performance metrics such as resource consumption, database objects, schema statistics and more, allowing you to proactively tune and fix issues in a High-Volume Production Environment.

[Toad](#) is a SQL Server DBMS toolset developed by Quest. It increases productivity by using extensive automation, intuitive workflows, and built-in expertise. This SQL management tool resolve issues, manage change and promote the highest levels of code quality for both relational and non-relational databases.

[Lepide SQL Server](#) is an open source storage manager utility to analyse the performance of SQL Servers. It provides a complete overview of all configuration and permission changes being made to your SQL Server environment through an easy-to-use, graphical user interface.

[Sequel Pro](#) is a fast MacOS database management tool for working with MySQL. This SQL management tool helpful for interacting with your database by easily to adding new databases, new tables, and new rows.

13. HMTL/CSS

[Back to the Top](#)



HTML/CSS Guide

A guide covering the HMTL/CSS languages including the applications and tools that will make you a better and more efficient HMTL/CSS developer.

Resources

[HTML \(HyperText Markup Language\)](#) is the basic building blocks of the Web. It defines the meaning and structure of web content along with other technologies used to describe a web page's appearance/presentation using CSS or functionality/behavior using JavaScript.

[Cascading Style Sheets \(CSS\)](#) is a stylesheet language used to describe the presentation of a document written in HTML or XML (including XML dialects such as SVG, MathML or XHTML). CSS also describes how elements should be rendered on screen, on paper, in speech, or on other media.

[Google HTML/CSS Style Guide](#)

[Airbnb CSS / Sass Style Guide](#)

[HTML Styles CSS](#)

[CSS Tutorial](#)

[Microsoft Certified Solutions Associate \(MCSA\): Web Applications](#)

[Intro to HTML/CSS: Making webpages by Khanacademy](#)

[Intro to HTML and CSS by Udacity](#)

[Learn to style HTML using CSS](#)

[The HTML and CSS Workshop](#)

[Modern HTML & CSS From The Beginning \(Including Sass\) by Udemy](#)

[Using Glitch for Developer Relations](#)

Tools

[WebStorm](#) is a professional IDE for JavaScript(including support for both HTML and CSS) developed by JetBrains. WebStorm comes with intelligent code completion, on-the-fly error detection, powerful navigation and refactoring for JavaScript, TypeScript, stylesheet languages, and all the most popular frameworks([Angular](#), [React](#), [Vue.js](#), [Ionic](#), [Apache Cordova](#), [React Native](#), [Node.js](#), [Meteor](#), and [Electron](#)).

[Codeanywhere](#) is a Cloud Integrated Development Environment. Our Cloud IDE saves you time by deploying a development environment in seconds, enabling you to code, learn, build, and collaborate on

your projects.

[Adobe Brackets](#) is a modern open-source code editor for HTML, CSS and JavaScript that's built in HTML, CSS and JavaScript.

[Ultimate CSS Gradient Generator](#) is a powerful Photoshop-like CSS gradient editor from [ColorZilla](#).

[CSS Gradient](#) is a happy little website and free tool that lets you create a gradient background for websites.

[Glitch](#) makes easier to build fast, full-stack web apps in your browser for free.

[CSS Optimizer](#) is an online tool to clean, compress, and optimize your CSS code.

[Sciter](#) is an embeddable HTML/CSS/scripting engine, Windows, MacOS and Linux.

[Flexy](#) is minimal CSS framework made with Flex.

[Simple CSS](#) is a tool that allows you to easily create Cascading Style Sheets from scratch, and modify existing ones, using a familiar point-and-click interface. With Simple CSS, you can manage multiple CSS projects, import existing .css files as desired, and export projects to .css files.

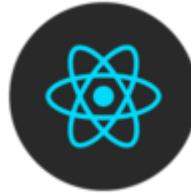
[Conditional-CSS](#) is a CSS module that allows to define a set of rules that will only apply based on the capabilities of the processor or the document the style sheet is being applied to.

[Kotatsu](#) is a straightforward CLI tool aiming either at running node.js scripts or serving JavaScript/TypeScript web applications in a modern environment.

[AWK](#) is a scripting language used for manipulating data and generating reports.

14. React

[Back to the Top](#)



ReactJS Guide

A guide covering ReactJS including the applications, libraries and tools that will make you a better and more efficient React developer.

Resources

[React](#) is a declarative, efficient, and flexible JavaScript library for building user interfaces.

[Getting Started with React](#)

[React JavaScript Tutorial in Visual Studio Code](#)

React Community Resources

[React Courses on Coursera](#)

[React Courses on Udemy](#)

[React Nanodegree program on Udacity](#)

[Becoming a React Developer Learning Path on LinkedIn Learning](#)

[Learning ReactJS on Codecademy](#)

[React Tutorials and Training Courses on Pluralsight](#)

[Introduction to React Course on Cloud Academy](#)

Tools

[WebStorm](#) is a professional IDE for JavaScript(including support for both HTML and CSS) developed by JetBrains. WebStorm comes with intelligent code completion, on-the-fly error detection, powerful navigation and refactoring for JavaScript, TypeScript, stylesheet languages, and all the most popular frameworks([Angular](#), [React](#), [Vue.js](#), [Ionic](#), [Apache Cordova](#), [React Native](#), [Node.js](#), [Meteor](#), and [Electron](#)).

[React Native](#) is a framework for building native apps for iOS and Android with React.

[Gatsby](#) is a free and open source framework based on React that helps developers build blazing fast websites and apps.

[React Starter Kit](#) is an isomorphic web app boilerplate for web development built on top of [Node.js](#), [Express](#), [GraphQL](#) and [React](#), containing modern web development tools such as [Webpack](#), [Babel](#) and [Browsersync](#). Helping you to stay productive following the best practices.

[React Hook Form](#) is a performant, flexible and extensible forms with easy to use validation(Web + React Native).

[GraphQL](#) is a query language for APIs and a runtime for fulfilling those queries with your existing data. It has support in Java, JavaScript, Ruby, Scala, and other programming languages.

[Apollo Client](#) is a fully-featured caching GraphQL client with integrations for React, Angular, and more. It allows you to easily build UI components that fetch data via GraphQL.

[Nest](#) is a framework for building efficient, scalable [Node.js](#) server-side applications. It uses modern JavaScript, is built with TypeScript (preserves compatibility with pure JavaScript) and combines elements of OOP (Object Oriented Programming), FP (Functional Programming), and FRP (Functional Reactive Programming).

[Meteor](#) is an ultra-simple environment for building modern web applications with JavaScript.

[mysqljs](#) is a pure node.js JavaScript Client implementing the MySQL protocol.

[axios](#) is a promise based HTTP client for the browser and node.js.

[Storybook](#) is a development environment for UI components. It allows you to browse a component library, view the different states of each component, and interactively develop and test components. It works with React, Vue, Angular, Ember, and other web frameworks.

[Next.js](#) is a React Framework for production gives you the best developer experience with all the features needed for production such as hybrid static & server rendering, TypeScript support, smart bundling, route pre-fetching, and more.

[React Boilerplate](#) is a highly scalable, offline-first foundation with the best developer experience and a focus on performance and best practices.

[TypeORM](#) is an ORM that can run in NodeJS, Browser, Cordova, PhoneGap, Ionic, React Native, NativeScript, Expo, and Electron platforms and can be used with TypeScript and JavaScript (ES5, ES6, ES7, ES8).

[Enzyme](#) is a JavaScript Testing utility for React that makes it easier to test your React Components' output. The user can also manipulate, traverse, and in some ways simulate runtime given the output.

[RxDB](#) is a NoSQL-database for JavaScript Applications like Websites, hybrid Apps, Electron-Apps, Progressive Web Apps and NodeJs.

[Redux](#) is a predictable state container for JavaScript apps.

[Inferno](#) is an insanely fast, React-like library for building high-performance user interfaces on both the client and server.

[Expo](#) is an open-source platform for making universal native apps with React.

[React Native Windows](#) is a framework for building native Windows apps with React. [React Native](#) is a framework developed by Facebook that enables you to build world-class application experiences on native platforms using a consistent developer experience based on JavaScript and React.

[ReactiveUI](#) is a composable, cross-platform model-view-viewmodel framework for all .NET platforms that is inspired by functional reactive programming, which is a paradigm that allows you to abstract mutable state away from your user interfaces and express the idea around a feature in one readable place and improve the testability of your application.

[Ant Design](#) is an enterprise-class UI design language and React UI library.

[Material-UI](#) is a collection of [React](#) components for faster and simpler web development.

[Chakra UI](#) is a set of accessible, reusable, and composable React components that make it super easy to create websites and apps.

15. Angular

[Back to the Top](#)



Angular Guide

A guide covering the Angular web framework including the applications and tools that will make you a better and more efficient Angular developer.

Resources

Angular is a development platform for building mobile and desktop web applications using Typescript/JavaScript and other languages developed by Google.

[Getting Started with Angular](#)

[What Is AngularJS?](#)

[Google's JavaScript Style Guide](#)

[Airbnb JavaScript Style Guide](#)

[AngularJS Tutorial on W3Schools](#)

[Angular Certification - Become a certified Angular Developer](#)

[Angular University](#)

[Angular Courses Online on Udemy](#)

[Angular Courses Online on Coursera](#)

[Learn Angular with Online Courses and Lessons on edX](#)

[Angular Online Training Courses on LinkedIn Learning](#)

[Learning AngularJS, including Angular directives from Codecademy](#)

[Learning Angular from Pluralsight](#)

[Ionic Angular Overview](#)

[Angular Language Service for Visual Studio](#)

[Angular JavaScript Tutorial in Visual Studio Code](#)

[Laravel and Angular](#)

Tools

Angular CLI is a command-line tool makes it easy to create an application that already works, right out of the box and follows all of Angular's best practices.

[Angular Material UI component library](#) is the Component infrastructure and Material Design components for Angular.

[WebStorm](#) is a professional IDE for JavaScript(including support for both HTML and CSS) developed by JetBrains. WebStorm comes with intelligent code completion, on-the-fly error detection, powerful navigation and refactoring for JavaScript, TypeScript, stylesheet languages, and all the most popular frameworks([Angular](#), [React](#), [Vue.js](#), [Ionic](#), [Apache Cordova](#), [React Native](#), [Node.js](#), [Meteor](#), and [Electron](#)).

[TypeORM](#) is an ORM that can run in NodeJS, Browser, Cordova, PhoneGap, Ionic, React Native, NativeScript, Expo, and Electron platforms and can be used with TypeScript and JavaScript (ES5, ES6, ES7, ES8).

[Storybook](#) is a development environment for UI components. It allows you to browse a component library, view the different states of each component, and interactively develop and test components. It works with React, Vue, Angular, Ember, and other web frameworks.

[AngularFire](#) is the official Angular library for [Firebase](#).

[Apollo Angular](#) is a GraphQL Client for Angular Framework.

[GraphQL Code Generator](#) is a tool for generating code based on a GraphQL schema and GraphQL operations. The project currently support and maintain [plugins](#) (TypeScript, Flow, React, Angular, MongoDB, Stencil, Reason, and more).

[NgRx](#) is the Reactive libraries for Angular.

[RxDB](#) is a NoSQL-database for JavaScript Applications like Websites, hybrid Apps, Electron-Apps, Progressive Web Apps and NodeJs.

[Redux](#) is a predictable state container for JavaScript apps.

[Protractor](#) is an end-to-end test framework for Angular and AngularJS applications. Protractor is a [Node.js](#) program built on top of [WebDriverJS](#).

[Onsen UI](#) is an open source framework that makes it easy to create native-feeling Progressive Web Apps (PWAs) and hybrid apps. It provides bindings for Angular 1, 2, React and Vue.js.

[Nebular](#) is a customizable Angular 10 UI Library with a focus on beautiful design and ability to adapt it to your brand easily. It comes with 4 visual themes, a powerful theming engine with runtime theme switching, and support of custom css properties mode.

[PrimeNG](#) is the Most Complete Angular UI Component Library.

[Project Clarity](#) is an open source design system that brings together UX guidelines, an HTML/CSS framework, Angular components, and Web Components. Project Clarity is currently being developed by VMware.

[Angular Flex Layout](#) is a sophisticated HTML UI layout for Angular applications; using Flexbox and a Responsive API.

[UI-Grid](#) is an AngularJS data grid.

16. VueJS

[Back to the Top](#)

VueJS Guide

A guide covering the VueJS web framework including the applications, libraries and tools that will make you a better and more efficient VueJS developer.

Resources

[Vue.js](#) is a progressive, incrementally-adoptable JavaScript framework for building UI on the web.

[Introduction to Vue.js](#)

[Vue.js API](#)

[Introduction to Vue Test Utils](#)

[TypeScript Support for Vue.js](#)

[Vue JavaScript Tutorial in Visual Studio Code](#)

[Vue.js Plugins for JetBrains](#)

[Introducing Storybook for VueJS](#)

[Vue Mastery is the Ultimate Learning Resource for Vue.js](#)

[Learn Vue.js from core-team members and industry experts on VueSchool](#)

[VueJS Courses on Udemy](#)

[Learning Vue.js on Codecademy](#)

[VueJS Gitter](#)

[VueJS Forum](#)

[VueJS Open Collective](#)

Tools

[Vue CLI](#) is a standard tooling for Vue.js development that provides support for Babel, TypeScript, ESLint, PostCSS, PWA, Unit Testing & End-to-end Testing.

[Vue Router](#) is the official router for Vue.js. It deeply integrates with Vue.js core to make building Single Page Applications with Vue.js a breeze.

[VuePress](#) is a tool that generates pre-rendered static HTML for each page, and runs as an SPA once a page is loaded.

[Vuefire](#) is Firebase bindings for Vue.js & Vuex.

[WebStorm](#) is a professional IDE for JavaScript(including support for both HTML and CSS) developed by JetBrains. WebStorm comes with intelligent code completion, on-the-fly error detection, powerful navigation and refactoring for JavaScript, TypeScript, stylesheet languages, and all the most popular frameworks([Angular](#), [React](#), [Vue.js](#), [Ionic](#), [Apache Cordova](#), [React Native](#), [Node.js](#), [Meteor](#), and [Electron](#)).

[Element](#) is a Vue.js 2.0 UI Toolkit for Web.

[Vuetify](#) is a Material Component Framework for Vue.

[Buefy](#) is a lightweight UI components for Vue.js based on Bulma.

[Storybook](#) is a development environment for UI components. It allows you to browse a component library, view the different states of each component, and interactively develop and test components. It works with React, Vue, Angular, Ember, and other web frameworks.

[TypeORM](#) is an ORM that can run in NodeJS, Browser, Cordova, PhoneGap, Ionic, React Native, NativeScript, Expo, and Electron platforms and can be used with TypeScript and JavaScript (ES5, ES6, ES7, ES8).

[Apollo Client](#) is a fully-featured caching GraphQL client with integrations for React, Angular, and more. It allows you to easily build UI components that fetch data via GraphQL.

[Vue-Apollo](#) is the Apollo/GraphQL integration for VueJS.

[RxDB](#) is a NoSQL-database for JavaScript Applications like Websites, hybrid Apps, Electron-Apps, Progressive Web Apps and NodeJs.

[Redux](#) is a predictable state container for JavaScript apps.

[Onsen UI](#) is an open source framework that makes it easy to create native-feeling Progressive Web Apps (PWAs) and hybrid apps. It provides bindings for Angular 1, 2, React and Vue.js.

[Nest](#) is a framework for building efficient, scalable [Node.js](#) server-side applications. It uses modern JavaScript, is built with TypeScript (preserves compatibility with pure JavaScript) and combines elements of OOP (Object Oriented Programming), FP (Functional Programming), and FRP (Functional Reactive Programming).

[Hoppscotch](#) is a free, fast and beautiful API request builder used by 100k+ developers.

[Quasar](#) is a framework that builds high-performance VueJS user interfaces in record time. Such as responsive Single Page Apps, SSR Apps, PWAs, Browser extensions, Hybrid Mobile Apps and Electron Apps.

17. Node.js

[Back to the Top](#)



Node.js Guide

A guide covering the Node.js including the applications and tools that will make you a better and more efficient Node.js developer.

Resources

[Node.js](#) is a JavaScript runtime built on Chrome's V8 JavaScript engine that lets developers write command line tools and server-side scripts outside of a browser.

[Node.js Build Working Group](#) maintains and controls infrastructure used for continuous integration (CI), releases, benchmarks, web hosting (of nodejs.org and other Node.js web properties) and more.

[The OpenJS Foundation](#) is made up of 32 open source JavaScript projects including Appium, Dojo, Electron, jQuery, Node.js, and webpack. The foundation's mission is to support the healthy growth of JavaScript and web technologies by providing a neutral organization to host and sustain projects, as well as collaboratively fund activities that benefit the ecosystem as a whole.

[Set up NodeJS on WSL 2](#)

[Getting started with Node.js in Google Cloud](#)

[Getting Started with Node.js in AWS](#)

[Node.js App Hosting & Deployment in Microsoft Azure](#)

[The Node.js best practices list](#)

[Introduction to Node.js by W3Schools](#)

[The Node.js Community Committee](#)

[Node.js Mentorship Program Initiative](#)

[Node.js tutorial in Visual Studio Code](#)

[Server-side Development with NodeJS, Express and MongoDB on Coursera](#)

Tools

[NPM](#) is the company behind Node package manager, the npm Registry, and npm CLI.

[node-gyp](#) is a cross-platform command-line tool written in Node.js for compiling native addon modules for Node.js. It contains a vendored copy of the gyp-next project that was previously used by the Chromium team, extended to support the development of Node.js native addons.

[nvm](#) is a version manager for node.js, designed to be installed per-user, and invoked per-shell. nvm works on any POSIX-compliant shell (sh, dash, ksh, zsh, bash), in particular on these platforms: unix, macOS, and windows WSL.

[node-docker](#) is the official Node.js docker image, made with love by the node community.

[Mocha](#) is a simple, flexible, fun JavaScript test framework for Node.js & The Browser.

[AVA](#) is a test runner for Node.js with a concise API, detailed error output, embrace of new language features and process isolation that lets you develop with confidence.

[egg](#) is a born to build better enterprise frameworks and apps with Node.js & Koa.

[mysqljs](#) is a pure node.js JavaScript Client implementing the MySQL protocol.

[axios](#) is a promise based HTTP client for the browser and node.js.

[Fastify](#) is a fast and low overhead web framework, for Node.js.

[Express](#) is a fast, unopinionated, minimalist web framework for node.

[Meteor](#) is an ultra-simple environment for building modern web applications with JavaScript.

[NW.js](#) is an app runtime based on Chromium and node.js. You can write native apps in HTML and JavaScript with NW.js. It also lets you call Node.js modules directly from the DOM and enables a new way of writing native applications with all Web technologies.

[PM2](#) is a production process manager for Node.js applications with a built-in load balancer. It allows you to keep applications alive forever, to reload them without downtime and to facilitate common system admin tasks.

[NestJS](#) is a framework for building efficient, scalable Node.js web applications. It uses modern JavaScript, is built with TypeScript and combines elements of OOP (Object Oriented Programming), FP (Functional Programming), and FRP (Functional Reactive Programming).

[jenkins-nodejs](#) is a Jenkins plugin for Node.js that provides the NodeJS auto-installer, allowing to create as many NodeJS installations "profiles" as you want.

[Strapi](#) is an open source Node.js Headless CMS to easily build customisable APIs.

[Standard](#) is a JavaScript Style Guide, with linter & automatic code fixer.

[React Starter Kit](#) is an isomorphic web app boilerplate for web development built on top of [Node.js](#), [Express](#), [GraphQL](#) and [React](#), containing modern web development tools such as [Webpack](#), [Babel](#) and [Browsersync](#). Helping you to stay productive following the best practices.

[Hexo](#) is a A fast, simple & powerful blog framework, powered by Node.js.

18. TypeScript

[Back to the Top](#)



TypeScript Guide

A guide covering the TypeScript programming language including the applications and tools that will make you a better and more efficient TypeScript developer.

Resources

TypeScript is a language for application-scale JavaScript. TypeScript adds optional types to JavaScript that support tools for large-scale JavaScript applications for any browser, for any host, on any OS. TypeScript compiles to readable, standards-based JavaScript.

[TypeScript support for Webpack](#)

[TypeScript Support for Nuxt.js](#)

[TypeScript Support for Vue.js](#)

[TypeScript Support for React Native](#)

[TypeScript Support for Angular](#)

[Ionic/TypeScript Starter Project](#)

[GitHub Actions for JavaScript and TypeScript](#)

Tools

[WebStorm](#) is a professional IDE for JavaScript(including support for both HTML and CSS) developed by JetBrains. WebStorm comes with intelligent code completion, on-the-fly error detection, powerful navigation and refactoring for JavaScript, TypeScript, stylesheet languages, and all the most popular frameworks([Angular](#), [React](#), [Vue.js](#), [Ionic](#), [Apache Cordova](#), [React Native](#), [Node.js](#), [Meteor](#), and [Electron](#)).

[ReSharper](#) is a Visual Studio Extension for .NET Developers. It comes with code quality analysis, which is available in C#, VB.NET, XAML, ASP.NET, ASP.NET MVC, JavaScript, TypeScript, CSS, HTML, and XML. You'll know right away if your code needs to be improved. ReSharper is one of many powerful tools developed by [JetBrains](#).

[ts-migrate](#) is a tool for helping migrate code to TypeScript. It takes a JavaScript, or a partial TypeScript, project in and gives a compiling TypeScript project out. ts-migrate is intended to accelerate the TypeScript migration process. The resulting code will pass the build, but a followup is required to improve type safety.

[Deno](#) is a simple, modern and secure runtime for JavaScript and TypeScript that uses V8 and is built in Rust.

[gulp-typescript](#) is an TypeScript compiler for gulp with incremental compilation support.

[React](#) is a declarative, efficient, and flexible JavaScript library for building user interfaces.

[React Native](#) is a framework for building native apps for iOS and Android with React.

[Vue.js](#) is a progressive, incrementally-adoptable JavaScript framework for building UI on the web.

[Angular](#) is a development platform for building mobile and desktop web applications using Typescript/JavaScript and other languages.

[Ionic Framework](#) is a powerful cross-platform UI toolkit for building native-quality iOS, Android, and Progressive Web Apps with HTML, CSS, and JavaScript.

[Stencil](#) is a simple compiler for generating Web Components and static site generated progressive web apps (PWA). Stencil was built by the Ionic team for its next generation of performant mobile and desktop Web Components.

[Storybook](#) is a development environment for UI components. It allows you to browse a component library, view the different states of each component, and interactively develop and test components. It works with React, Vue, Angular, Ember, and other web frameworks.

[Prettier](#) is an opinionated code formatter. It enforces a consistent style by parsing your code and re-printing it with its own rules that take the maximum line length into account, wrapping code when necessary.

[Nest](#) is a framework for building efficient, scalable [Node.js](#) server-side applications. It is built with TypeScript and combines elements of Object Oriented Programming(OOP), Functional Programming, and Functional Reactive Programming(FRP).

[Definitely Typed](#) is a repository for high quality TypeScript type definitions.

[TypeORM](#) is an ORM that can run in NodeJS, Browser, Cordova, PhoneGap, Ionic, React Native, NativeScript, Expo, and Electron platforms and can be used with TypeScript and JavaScript (ES5, ES6, ES7, ES8).

[NativeScript](#) empowers you to access native APIs from JavaScript directly. The framework currently provides iOS and Android runtimes for rich mobile development and can be utilized in a number of diverse use cases.

[AssemblyScript](#) compiles a strict variant of TypeScript to [WebAssembly](#) using [Binaryen](#).

[React Hook Form](#) is a performant, flexible and extensible forms with easy to use validation(Web + React Native).

[Apollo Client](#) is a fully-featured caching GraphQL client with integrations for React, Angular, and more. It allows you to easily build UI components that fetch data via GraphQL.

[TensorFlow.js](#) is an open-source WebGL hardware-accelerated JavaScript library for training and deploying machine learning models.

[Rome](#) is a linter, compiler, bundler, and [more](#) for JavaScript, TypeScript, JSON, HTML, Markdown, and CSS.

[Eclipse Theia](#) is an extensible platform to develop full-fledged multi-language Cloud & Desktop IDE-like products with state-of-the-art web technologies.

[InversifyJS](#) is a powerful and lightweight inversion of control(IoC) container for JavaScript & Node.js apps powered by TypeScript. An IoC container uses a class constructor to identify and inject its dependencies.

[Gatsby](#) is a free and open source framework based on React that helps developers build blazing fast websites and apps.

[Meteor](#) is an ultra-simple environment for building modern web applications with JavaScript.

[TypeScript ESLint](#) is a monorepo for all the tooling which enables ESLint to support TypeScript.

[TS node](#) is a TypeScript execution and REPL for node.js.

19. Rust

[Back to the Top](#)



Rust Guide

A guide covering the Rust programming language including the applications and tools that will make you a better and more efficient Rust developer.

Resources

Rust is a multi-paradigm programming language focused on performance and safety. Rust has a comparable amount of runtime to C and C++, and has set up its standard library to be amenable towards OS development. Specifically, the standard library is split into two parts: core and std. Core is the lowest-level aspects only, and doesn't include things like allocation, threading, and other higher-level features.

[The Rust Language Reference](#)

[The Rust Programming Language Book](#)

[Learning Rust](#)

[Why AWS loves Rust](#)

[Rust Programming courses on Udemy](#)

[Safety in Systems Programming with Rust at Standford by Ryan Eberhardt](#)

[WebAssembly meets Kubernetes with Krustlet using Rust](#)

[Microsoft's Project Verona](#)

Tools

[Cargo](#) is a package manager that downloads your Rust project's dependencies and compiles your project.

[Crater](#) is a tool to run experiments across parts of the Rust ecosystem. Its primary purpose is to detect regressions in the Rust compiler, and it does this by building a large number of crates, running their test suites and comparing the results between two versions of the Rust compiler. It can operate locally (with Docker as the only dependency) or distributed on the cloud. It can operate locally (with Docker as the only dependency) or distributed on the cloud.

[VSCode-Rust](#) is plugin that adds language support for Rust to Visual Studio Code. Rust support is powered by a separate language server - either by the official Rust Language Server (RLS) or rust-analyzer, depending on the user's preference. If you don't have it installed, the extension will install it for you (with permission). This extension is built and maintained by the Rust IDEs and editors team with the focus on providing a stable, high quality extension that makes the best use of the respective language server.

[Apache Arrow](#) is a development platform for in-memory analytics. It contains a set of technologies that enable big data systems to process and move data fast. Arrow's libraries are available for C, C++, C#, Go, Java, JavaScript, MATLAB, Python, R, Ruby, and Rust.

[Wasmer](#) enables super lightweight containers based on [WebAssembly](#) that can run anywhere such as the Desktop to the Cloud and IoT devices, and also embedded in [any programming language](#).

[Firecracker](#) is an open source virtualization technology that is purpose-built for creating and managing secure, multi-tenant container and function-based services that provide serverless operational models. Firecracker runs workloads in lightweight virtual machines, called microVMs, which combine the security and isolation properties provided by hardware virtualization technology with the speed and flexibility of containers. Firecracker has also been integrated in container runtimes, for example [Kata Containers](#) and [Weaveworks Ignite](#).

[Tokio](#) is an event-driven, non-blocking I/O platform for writing asynchronous applications with the Rust programming language.

[TiKV](#) is an open-source distributed transactional key-value database that also provides classical key-value APIs, but also transactional APIs with ACID compliance.

[Sonic](#) is a fast, lightweight and schema-less search backend similar to Elasticsearch in some use-cases.

[Hyper](#) is a fast and correct HTTP library for Rust.

[Rocket](#) is an async web framework for Rust with a focus on usability, security, extensibility, and speed.

[Clippy](#) is a collection of lints to catch common mistakes and improve your Rust code.

[Servo](#) is a prototype web browser engine written in the Rust language.

[Vector](#) is a high-performance, end-to-end (agent & aggregator) observability data platform that puts the user in control of their observability data.

[RustPython](#) is a Python Interpreter written in Rust.

[Miri](#) is an interpreter for Rust's mid-level intermediate representation. It can run binaries and test suites of cargo projects and detect certain classes of undefined behavior. Miri will also tell you about

memory leaks: when there is memory still allocated at the end of the execution, and that memory is not reachable from a global static, Miri will raise an error.

[Chalk](#) is an implementation and definition of the Rust trait system using a PROLOG-like logic solver.

[stdarch](#) is Rust's standard library vendor-specific APIs and run-time feature detection.

[Simpleinfra](#) is rep that contains the tools and automation written by the Rust infrastructure team to manage our services. Using some of the tools in this repo require privileges only infra team members have.

[Rustlings](#) is a small set of exercises to get you used to reading and writing Rust code.

[Krustlet](#) acts as a Kubernetes Kubelet(written in Rust) by listening on the event stream for new pods that the scheduler assigns to it based on specific Kubernetes [tolerations](#). The project is currently experimental.

Operating System

[Redox](#) is a Unix-like Operating System written in Rust, aiming to bring the innovations of Rust to a modern microkernel and full set of applications. Actively being developed by [Jeremy Soeller](#).

[Bottlerocket OS](#) is an open-source Linux-based operating system meant for hosting containers.

Bottlerocket focuses on security and maintainability, providing a reliable, consistent, and safe platform for container-based workloads.

[Tock](#) is an embedded operating system designed for running multiple concurrent, mutually distrustful applications on Cortex-M and RISC-V based embedded platforms. Tock's design centers around protection, both from potentially malicious applications and from device drivers. Tock uses two mechanisms to protect different components of the operating system. First, the kernel and device drivers are written in Rust, a systems programming language that provides compile-time memory safety, type safety and strict aliasing. Tock uses Rust to protect the kernel (the scheduler and hardware abstraction layer) from platform specific device drivers as well as isolate device drivers from each other. Second, Tock uses memory protection units to isolate applications from each other and the kernel.

[Rust on Chrome OS](#) is a document that provides information on creating Rust projects for installation within Chrome OS and Chrome OS SDK.

[Writing an OS in Rust](#) is a blog series creates a small operating system in the Rust programming language by [Philipp Oppermann](#).

20. Kotlin

[Back to the Top](#)



Kotlin Guide

A guide covering the Kotlin programming language including the applications and tools that will make you a better and more efficient Kotlin developer.

Resources

[Kotlin](#) is a very popular programming language that targets the JVM. Kotlin has experienced a surge in popularity the last few years making it the most popular JVM language not including Java.

[Kotlin Programming Language Reference](#)

[Google Developers Training](#)

[Google Developers Certification](#)

[Kotlin style guide](#)

[Learn the Kotlin programming language](#)

[Get Started with Kotlin on Android](#)

[Kotlin for cross-platform mobile development](#)

[Using Kotlin with Quakus](#)

[Building web applications with Spring Boot and Kotlin](#)

[Migrating build logic from Groovy to Kotlin](#)

[Kotlin Playground: Edit, Run, Share Kotlin Code Online](#)

[An Absolute Beginner's Guide to Kotlin by Treehouse](#)

[Kotlin Courses Online from Udemy](#)

[Learning Kotlin from Codecademy](#)

[Kotlin Bootcamp for Programmers course from Udacity](#)

Tools

[Android Studio](#) is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Available on Windows, macOS, Linux, and Chrome OS.

[IntelliJ IDEA](#) is an IDE for Java, but it also understands and provides intelligent coding assistance for a large variety of other languages such as Kotlin, SQL, JPQL, HTML, JavaScript, etc., even if the language expression is injected into a String literal in your Java code.

[Gradle](#) is a build automation tool that supports multi-language development. If you're building, testing, publishing, and deploying software on any platform, Gradle offers a flexible model that can support the entire development lifecycle from compiling and packaging code to publishing web sites. Gradle is designed to support build automation across multiple languages and platforms including Java, Scala,

Android, C/C++, Swift, and Groovy, which is closely integrated with development tools and continuous integration servers including Eclipse, IntelliJ, and Jenkins.

[vue-kotlin](#) is libraries and tools supporting the use of Vue.js in Kotlin.

[kotlinx-lincheck](#) is a framework for testing concurrent data structures for correctness.

[kotlinx-io](#) is a multiplatform library for processing binary data, working with memory blocks, interacting with the platform, and performing other low level operations.

[kotlinx-knit](#) is a tool that produces Kotlin source example files and tests from markdown documents with embedded snippets of Kotlin code. It also helps to add links to the API documentation website into the documents and has a few other helpful markdown-management features.

[Kotlin Xcode compatibility Gradle plugin](#) is a plugin used by [AppCode](#) to set up Kotlin/Native project along with Xcode.

21. Swift

[Back to the Top](#)



Swift Guide

A guide covering the Swift programming language including the applications and tools that will make you a better and more efficient Swift developer.

Resources

[Swift](#) is Apple's main programming language for iOS, macOS, watchOS, and tvOS app development. Though, many parts of Swift will be familiar to developers from their experience of developing in C and Objective-C.

[Swift Evolution](#) maintains proposals for changes and user-visible enhancements to the Swift Programming Language.

[Xcode + Swift](#) makes developing applications for MacOS and iOS fast and fun.

Swift 5.3 Basics

[Start Developing iOS Apps with Swift](#)

[Apple Developer Documentation](#)

[Apple Foundation Framework](#)

[Apple Core Animation Framework](#)

[Apple Core Graphics Framework](#)

[Getting Started with LLDB](#)

[Mac Catalyst - iOS - Human Interface Guidelines](#)

[Amazon EC2 Mac Instances](#)

[Swift GitHub](#)

[Apple Developer Forums](#)

[Swift Forums](#)

[Google's Swift Style Guide](#)

[Swift Courses Online from Coursera](#)

[Swift Courses Online from Udemy](#)

[Learning Swift course from Codecademy](#)

Tools

[Xcode](#) includes everything developers need to create great applications for Mac, iPhone, iPad, Apple TV, and Apple Watch. Xcode provides developers a unified workflow for user interface design, coding, testing, and debugging. Xcode 12 is built as an Universal app that runs 100% natively on Intel-based CPUs and Apple Silicon. It includes a unified macOS SDK that features all the frameworks, compilers, debuggers, and other tools you need to build apps that run natively on Apple Silicon and the Intel x86_64 CPU.

[SwiftUI](#) is a user interface toolkit that provides views, controls, and layout structures for declaring your app's user interface. The SwiftUI framework provides event handlers for delivering taps, gestures, and other types of input to your application.

[UIKit](#) is a framework provides the required infrastructure for your iOS or tvOS apps. It provides the window and view architecture for implementing your interface, the event handling infrastructure for delivering Multi-Touch and other types of input to your app, and the main run loop needed to manage interactions among the user, the system, and your app.

[AppKit](#) is a graphical user interface toolkit that contains all the objects you need to implement the user interface for a macOS app such as windows, panels, buttons, menus, scrollers, and text fields, and it handles all the details for you as it efficiently draws on the screen, communicates with hardware devices and screen buffers, clears areas of the screen before drawing, and clips views.

[ARKit](#) is a set set of software development tools to enable developers to build augmented-reality apps for iOS developed by Apple. The latest version ARKit 3.5 takes advantage of the new LiDAR Scanner and depth sensing system on iPad Pro(2020) to support a new generation of AR apps that use Scene Geometry for enhanced scene understanding and object occlusion.

[RealityKit](#) is a framework to implement high-performance 3D simulation and rendering with information provided by the ARKit framework to seamlessly integrate virtual objects into the real world.

[SceneKit](#) is a high-level 3D graphics framework that helps you create 3D animated scenes and effects in your iOS apps.

[Mac Catalyst](#) is a set of Apple APIs that developers can use to rapidly port their iOS apps to [Apple Silicon M1 Chip](#) and take full advantage of the new capabilities on the new Apple hardware.

[Instruments](#) is a powerful and flexible performance-analysis and testing tool that's part of the Xcode tool set. It's designed to help you profile your iOS, watchOS, tvOS, and macOS apps, processes, and devices in order to better understand and optimize their behavior and performance.

[Cocoapods](#) is a dependency manager for Swift and Objective-C used in Xcode projects by specifying the dependencies for your project in a simple text file. CocoaPods then recursively resolves dependencies between libraries, fetches source code for all dependencies, and creates and maintains an Xcode workspace to build your project.

[AppCode](#) is constantly monitoring the quality of your code. It warns you of errors and smells and suggests quick-fixes to resolve them automatically. AppCode provides lots of code inspections for Objective-C, Swift, C/C++, and a number of code inspections for other supported languages.

[Vapor](#) is a web framework for Swift. It provides a beautifully expressive and easy to use foundation for your next website, API, or cloud project.

[Hero](#) is a library for building iOS view controller transitions. It provides a declarative layer on top of the UIKit's cumbersome transition APIs—making custom transitions an easy task for developers.

[Kingfisher](#) is a powerful, pure-Swift library for downloading and caching images from the web. It provides you a chance to use a pure-Swift way to work with remote images in your next app.

[Realm](#) is a mobile database that runs directly inside phones, tablets or wearables. This repository holds the source code for the iOS, macOS, tvOS & watchOS versions of Realm Swift & Realm Objective-C.

[Perfect](#) is a complete and powerful toolbox, framework, and application server for Linux, iOS, and macOS (OS X). It provides everything a Swift engineer needs for developing lightweight, maintainable, and scalable apps and other REST services entirely in the Swift programming language for both client-facing and server-side applications.

[Alamofire](#) is an HTTP networking library written in Swift.

[Eureka](#) is an elegant iOS form builder in Swift

[Carthage](#) is intended to be the simplest way to add frameworks to your Cocoa application. Carthage builds your dependencies and provides you with binary frameworks, but you retain full control over your project structure and setup. Carthage does not automatically modify your project files or your build settings.

[ReactiveCocoa](#) is reactive extensions to Cocoa frameworks, built on top of ReactiveSwift.

22. Golang

[Back to the Top](#)



Go Guide

A guide covering the Go programming language including the applications and tools that will make you a better and more efficient Go developer.

Resources

Go is an open source programming language that makes it easy to build simple, reliable, and efficient software.

[Golang Contribution Guide](#)

[Google Developers Training](#)

[Google Developers Certification](#)

[Uber's Go Style Guide](#)

[GitLab's Go standards and style guidelines](#)

[Effective Go](#)

[Go: The Complete Developer's Guide \(Golang\) on Udemy](#)

[Getting Started with Go on Coursera](#)

[Programming with Google Go on Coursera](#)

[Learning Go Fundamentals on Pluralsight](#)

[Learning Go on Codecademy](#)

Tools

[golang tools](#) holds the source for various packages and tools that support the Go programming language.

[Go in Visual Studio Code](#) is an extension that gives you language features like IntelliSense, code navigation, symbol search, bracket matching, snippets, and many more that will help you in Golang development.

[Traefik](#) is a modern HTTP reverse proxy and load balancer that makes deploying microservices easy. Traefik integrates with your existing infrastructure components (Docker, Swarm mode, Kubernetes, Marathon, Consul, Etcd, Rancher, Amazon ECS, etc.) and configures itself automatically and dynamically. Pointing Traefik at your orchestrator should be the only configuration step you need.

[Gitea](#) is Git with a cup of tea, painless self-hosted git service. Using Go, this can be done with an independent binary distribution across all platforms which Go supports, including Linux, macOS, and Windows on x86, amd64, ARM and PowerPC architectures.

[OpenFaaS](#) is Serverless Functions Made Simple. It makes it easy for developers to deploy event-driven functions and microservices to Kubernetes without repetitive, boiler-plate coding. Package your code or an existing binary in a Docker image to get a highly scalable endpoint with auto-scaling and metrics.

[micro](#) is a terminal-based text editor that aims to be easy to use and intuitive, while also taking advantage of the capabilities of modern terminals. As its name indicates, micro aims to be somewhat of a successor to the nano editor by being easy to install and use. It strives to be enjoyable as a full-time editor for people who prefer to work in a terminal, or those who regularly edit files over SSH.

[Gravitational Teleport](#) is a modern security gateway for remotely accessing into Clusters of Linux servers via SSH or SSH-over-HTTPS in a browser or Kubernetes clusters.

[NATS](#) is a simple, secure and performant communications system for digital systems, services and devices. NATS is part of the Cloud Native Computing Foundation (CNCF). NATS has over 30 client language implementations, and its server can run on-premise, in the cloud, at the edge, and even on a Raspberry Pi. NATS can secure and simplify design and operation of modern distributed systems.

[Act](#) is a GO program that allows you to run our GitHub Actions locally.

[Fiber](#) is an [Express](#) inspired web framework built on top of [Fasthttp](#), the fastest HTTP engine for Go. Designed to ease things up for fast development with zero memory allocation and performance in mind.

[Glide](#) is a vendor Package Management for Golang.

[BadgerDB](#) is an embeddable, persistent and fast key-value (KV) database written in pure Go. It is the underlying database for [Dgraph](#), a fast, distributed graph database. It's meant to be a performant alternative to non-Go-based key-value stores like RocksDB.

[Go kit](#) is a programming toolkit for building microservices (or elegant monoliths) in Go. We solve common problems in distributed systems and application architecture so you can focus on delivering business value.

[Codis](#) is a proxy based high performance Redis cluster solution written in Go.

[zap](#) is a blazing fast, structured, leveled logging in Go.

[HttpRouter](#) is a lightweight high performance HTTP request router (also called multiplexer or just mux for short) for Go.

[Gorilla WebSocket](#) is a Go implementation of the WebSocket protocol.

[Delve](#) is a debugger for the Go programming language.

[GORM](#) is a fantastic ORM library for Golang, aims to be developer friendly.

[Go Patterns](#) is a curated collection of idiomatic design & application patterns for Go language.

23. C++

[Back to the Top](#)



C++ Guide

A guide covering C++ programming language including the applications and tools that will make you a better and more efficient C/C++ developer.

Resources

C++ is a cross-platform language that can be used to build high-performance applications developed by Bjarne Stroustrup, as an extension to the C language.

[Google C++ Style Guide](#)

[C++ style guide for Fuchsia](#)

[C and C++ Coding Style Guide by OpenTitan](#)

[Chromium C++ Style Guide](#)

[C++ Core Guidelines](#)

[C++ Style Guide for ROS](#)

[Learn C Programming Online Courses by edX](#)

[Learn C++ with Online Courses by edX](#)

[Learn C++ by Codecademy](#)

[C++ For C Programmers by Coursera](#)

Tools

[Visual Studio](#) is an integrated development environment (IDE) from Microsoft; which is a feature-rich application that can be used for many aspects of software development. Visual Studio makes it easy to edit, debug, build, and publish your app. By using Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, and Windows Store.

[Vcpkg](#) is a C++ Library Manager for Windows, Linux, and MacOS.

[ReSharper C++](#) is a Visual Studio Extension for C++ developers developed by JetBrains.

[CLion](#) is a cross-platform IDE for C and C++ developers developed by JetBrains.

[libcu++](#) is the NVIDIA C++ Standard Library for your entire system. It provides a heterogeneous implementation of the C++ Standard Library that can be used in and between CPU and GPU code.

[Thrust](#) is a C++ parallel programming library which resembles the C++ Standard Library. Thrust's high-level interface greatly enhances programmer productivity while enabling performance portability between GPUs and multicore CPUs. Interoperability with established technologies such as CUDA, TBB, and OpenMP integrates with existing software.

[ANTLR \(ANother Tool for Language Recognition\)](#) is a powerful parser generator for reading, processing, executing, or translating structured text or binary files. It's widely used to build languages, tools, and frameworks. From a grammar, ANTLR generates a parser that can build parse trees and also generates a listener interface that makes it easy to respond to the recognition of phrases of interest.

[Oat++](#) is a light and powerful C++ web framework for highly scalable and resource-efficient web application. It's zero-dependency and easy-portable.

[JavaCPP](#) is a program that provides efficient access to native C++ inside Java, not unlike the way some C/C++ compilers interact with assembly language.

[Cython](#) is a language that makes writing C extensions for Python as easy as Python itself. Cython is based on Pyrex, but supports more cutting edge functionality and optimizations such as calling C functions and declaring C types on variables and class attributes.

[Spdlog](#) is a very fast, header-only/compiled, C++ logging library.

[Infer](#) is a static analysis tool for Java, C++, Objective-C, and C. Infer is written in OCaml.

24. Ruby

[Back to the Top](#)



Ruby Guide

A guide covering the Ruby programming language including the applications and tools that will make you a better and more efficient Ruby developer.

Resources

[Ruby](#) is a dynamic, open source programming language with a focus on simplicity and productivity. It has an elegant syntax that is natural to read and easy to write.

[Ruby Documentation](#)

[Ruby Community](#)

[Ruby Gems](#)

[Ruby courses by Coursera](#)

[Learn Ruby course by Codecademy](#)

[Ruby Glossary](#)

[Ruby in Twenty Minutes Quickstart](#)

[Getting started with a Ruby on Rails application on CircleCI.](#)

[The Ruby Style Guide](#)

[Airbnb's Ruby Style Guide](#)

Tools

[RubyMine](#) is a professional IDE developed by Jet Brains that provides support for Ruby, Ruby on Rails and web development.

[Rails](#) is a web-application framework that includes everything needed to create database-backed web applications according to the [Model-View-Controller \(MVC\)](#) pattern. Understanding the MVC pattern is key to understanding Rails. MVC divides your application into three layers: Model, View, and Controller, each with a specific responsibility.

[rbenv](#) allows to pick a Ruby version for your application and guarantee that your development environment matches production. Put rbenv to work with Bundler for painless Ruby upgrades and bulletproof deployments.

[Prettier for Ruby](#) is a plugin for the Ruby programming language and its ecosystem. prettier is an opinionated code formatter that supports multiple languages and integrates with most editors. The idea is to eliminate discussions of style in code review and allow developers to get back to thinking about code design instead.

[Active Admin](#) is a Ruby on Rails framework for creating elegant backends for website administration.

[Capistrano](#) is a framework for building automated deployment scripts. Although Capistrano itself is written in Ruby, it can easily be used to deploy projects of any language or framework, be it Rails, Java, or PHP.

[Spree](#) is an open source E-commerce platform for Rails 6 with a modern UX, optional PWA frontend, REST API, GraphQL, several official extensions and 3rd party integrations.

[Sidekiq](#) is a simple, efficient background processing for Ruby. It uses hreads to handle many jobs at the same time in the same process. It does not require Rails but will integrate tightly with Rails to make background processing dead simple.

[Kaminari](#) is a Scope and Engine based, clean, powerful, and customizable paginator for modern web app frameworks and ORMs.

[React-Rails](#) is a flexible tool to use [React](#) with Rails. By integrating React.js with Rails views and controllers, the asset pipeline, or webpacker.

[Pry](#) is a runtime developer console and IRB alternative with powerful introspection capabilities.

[Brakeman](#) is a static analysis tool which checks Ruby on Rails applications for security vulnerabilities.

[dotenv](#) is a Ruby gem to load environment variables from `.env`.

[Scientist](#) is a Ruby library for carefully refactoring critical paths.

[Fastlane](#) is a tool written in Ruby for iOS and Android developers to automate tedious tasks like generating screenshots, dealing with provisioning profiles, and releasing your application.

[Fluentd](#) collects events from various data sources and writes them to files, RDBMS, NoSQL, IaaS, SaaS, Hadoop and so on all written in Ruby.

25. Java

[Back to the Top](#)



Java Guide

A guide covering Java including the applications and tools that will make you a better and more efficient Java developer.

Resources

[Java](#) is a popular programming language and development platform(JDK). It reduces costs, shortens development timeframes, drives innovation, and improves application services. With millions of developers running more than 51 billion Java Virtual Machines worldwide.

[The Eclipse Foundation](#) is home to a worldwide community of developers, the Eclipse IDE, Jakarta EE and over 375 open source projects, including runtimes, tools and frameworks for Java and other languages.

[Getting Started with Java](#)

[Oracle Java certifications from Oracle University](#)

[Google Developers Training](#)

[Google Developers Certification](#)

[Java Tutorial by W3Schools](#)

[Building Your First Android App in Java](#)

[Getting Started with Java in Visual Studio Code](#)

[Google Java Style Guide](#)

[AOSP Java Code Style for Contributors](#)

[Chromium Java style guide](#)

[Get Started with OR-Tools for Java](#)

[Getting started with Java Tool Installer task for Azure Pipelines](#)

[Gradle User Manual](#)

Tools

[Java SE](#) contains several tools to assist in program development and debugging, and in the monitoring and troubleshooting of production applications.

[JDK Development Tools](#) includes the Java Web Start Tools (javaws) Java Troubleshooting, Profiling, Monitoring and Management Tools (jcmd, jconsole, jmc, jvisualvm); and Java Web Services Tools (schemagen, wsgen, wsimport, xjc).

[Android Studio](#) is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Available on Windows, macOS, Linux, Chrome OS.

[IntelliJ IDEA](#) is an IDE for Java, but it also understands and provides intelligent coding assistance for a large variety of other languages such as Kotlin, SQL, JPQL, HTML, JavaScript, etc., even if the language expression is injected into a String literal in your Java code.

[NetBeans](#) is an IDE provides Java developers with all the tools needed to create professional desktop, mobile and enterprise applications. Creating, Editing, and Refactoring. The IDE provides wizards and templates to let you create Java EE, Java SE, and Java ME applications.

[Java Design Patterns](#) is a collection of the best formalized practices a programmer can use to solve common problems when designing an application or system.

[Elasticsearch](#) is a distributed RESTful search engine built for the cloud written in Java.

[RxJava](#) is a Java VM implementation of [Reactive Extensions](#): a library for composing asynchronous and event-based programs by using observable sequences. It extends the [observer pattern](#) to support sequences of data/events and adds operators that allow you to compose sequences together declaratively while abstracting away concerns about things like low-level threading, synchronization, thread-safety and concurrent data structures.

[Guava](#) is a set of core Java libraries from Google that includes new collection types (such as multimap and multiset), immutable collections, a graph library, and utilities for concurrency, I/O, hashing, caching, primitives, strings, and more! It is widely used on most Java projects within Google, and widely used by many other companies as well.

[okhttp](#) is a HTTP client for Java and Kotlin developed by Square.

[Retrofit](#) is a type-safe HTTP client for Android and Java developed by Square.

[LeakCanary](#) is a memory leak detection library for Android developed by Square.

[Apache Spark](#) is a unified analytics engine for large-scale data processing. It provides high-level APIs in Scala, Java, Python, and R, and an optimized engine that supports general computation graphs for data analysis. It also supports a rich set of higher-level tools including Spark SQL for SQL and DataFrames, MLlib for machine learning, GraphX for graph processing, and Structured Streaming for stream processing.

[Apache Flink](#) is an open source stream processing framework with powerful stream- and batch-processing capabilities with elegant and fluent APIs in Java and Scala.

[Fastjson](#) is a Java library that can be used to convert Java Objects into their JSON representation. It can also be used to convert a JSON string to an equivalent Java object.

[libGDX](#) is a cross-platform Java game development framework based on OpenGL (ES) that works on Windows, Linux, Mac OS X, Android, your WebGL enabled browser and iOS.

[Jenkins](#) is the leading open-source automation server. Built with Java, it provides over 1700 [plugins](#) to support automating virtually anything, so that humans can actually spend their time doing things machines cannot.

[DBeaver](#) is a free multi-platform database tool for developers, SQL programmers, database administrators and analysts. Supports any database which has JDBC driver (which basically means - ANY database). EE version also supports non-JDBC datasources (MongoDB, Cassandra, Redis, DynamoDB, etc).

[Redisson](#) is a Redis Java client with features of In-Memory Data Grid. Over 50 Redis based Java objects and services: Set, Multimap, SortedSet, Map, List, Queue, Deque, Semaphore, Lock, AtomicLong, Map Reduce, Publish / Subscribe, Bloom filter, Spring Cache, Tomcat, Scheduler, JCache API, Hibernate, MyBatis, RPC, and local cache.

[GraalVM](#) is a universal virtual machine for running applications written in JavaScript, Python, Ruby, R, JVM-based languages like Java, Scala, Clojure, Kotlin, and LLVM-based languages such as C and C++.

[Gradle](#) is a build automation tool for multi-language software development. From mobile apps to microservices, from small startups to big enterprises, Gradle helps teams build, automate and deliver better software, faster. Write in Java, C++, Python or your language of choice.

[Apache Groovy](#) is a powerful, optionally typed and dynamic language, with static-typing and static compilation capabilities, for the Java platform aimed at improving developer productivity thanks to a concise, familiar and easy to learn syntax. It integrates smoothly with any Java program, and immediately delivers to your application powerful features, including scripting capabilities, Domain-Specific Language authoring, runtime and compile-time meta-programming and functional programming.

[JaCoCo](#) is a free code coverage library for Java, which has been created by the EclEmma team based on the lessons learned from using and integration existing libraries for many years.

[Apache JMeter](#) is used to test performance both on static and dynamic resources, Web dynamic applications. It also used to simulate a heavy load on a server, group of servers, network or object to test its strength or to analyze overall performance under different load types.

[JUnit](#) is a simple framework to write repeatable tests. It is an instance of the xUnit architecture for unit testing frameworks.

[Mockito](#) is the most popular Mocking framework for unit tests written in Java.

[SpotBugs](#) is a program which uses static analysis to look for bugs in Java code.

[SpringBoot](#) is a great tool that helps you to create Spring-powered, production-grade applications and services with absolute minimum fuss. It takes an opinionated view of the Spring platform so that new and existing users can quickly get to the bits they need.

[YourKit](#) is a technology leader, creator of the most innovative and intelligent tools for profiling Java & .NET applications.

26. Scala

[Back to the Top](#)



Scala Guide

A guide covering Scala language including the applications and tools that will make you a better and more efficient Scala developer.

Resources

[Scala](#) is a combination of object-oriented and functional programming in one concise, high-level language. Scala's static types help avoid bugs in complex applications, and its JVM and JavaScript runtimes let you build high-performance systems with easy access to huge ecosystems of libraries.

[Scala Style Guide](#)

[Databricks Scala Style Guide](#)

[Data Science using Scala and Spark on Azure](#)

[Creating a Scala Maven application for Apache Spark in HDInsight using IntelliJ](#)

[Intro to Spark DataFrames using Scala with Azure Databricks](#)

[Using Scala to Program AWS Glue ETL Scripts](#)

[Using Flink Scala shell with Amazon EMR clusters](#)

[AWS EMR and Spark 2 using Scala from Udemy](#)

[Using the Google Cloud Storage connector with Apache Spark](#)

[Write and run Spark Scala jobs on Cloud Dataproc for Google Cloud](#)

[Scala Courses and Certifications from edX](#)

[Scala Courses from Coursera](#)

[Top Scala Courses from Udemy](#)

Tools

[Apache Spark](#) is a unified analytics engine for large-scale data processing. It provides high-level APIs in Scala, Java, Python, and R, and an optimized engine that supports general computation graphs for data analysis. It also supports a rich set of higher-level tools including Spark SQL for SQL and DataFrames, MLLib for machine learning, GraphX for graph processing, and Structured Streaming for stream processing.

[Apache Spark Connector for SQL Server and Azure SQL](#) is a high-performance connector that enables you to use transactional data in big data analytics and persists results for ad-hoc queries or reporting. The connector allows you to use any SQL database, on-premises or in the cloud, as an input data source or output data sink for Spark jobs.

[Azure Databricks](#) is a fast and collaborative Apache Spark-based big data analytics service designed for data science and data engineering. Azure Databricks, sets up your Apache Spark environment in minutes, autoscale, and collaborate on shared projects in an interactive workspace. Azure Databricks supports Python, Scala, R, Java, and SQL, as well as data science frameworks and libraries including TensorFlow, PyTorch, and scikit-learn.

[Apache PredictionIO](#) is an open source machine learning framework for developers, data scientists, and end users. It supports event collection, deployment of algorithms, evaluation, querying predictive results via REST APIs. It is based on scalable open source services like Hadoop, HBase (and other DBs), Elasticsearch, Spark and implements what is called a Lambda Architecture.

[Cluster Manager for Apache Kafka\(CMAK\)](#) is a tool for managing [Apache Kafka](#) clusters.

[BigDL](#) is a distributed deep learning library for Apache Spark. With BigDL, users can write their deep learning applications as standard Spark programs, which can directly run on top of existing Spark or Hadoop clusters.

[Eclipse Deeplearning4J \(DL4J\)](#) is a set of projects intended to support all the needs of a JVM-based(Scala, Kotlin, Clojure, and Groovy) deep learning application. This means starting with the raw data, loading and preprocessing it from wherever and whatever format it is in to building and tuning a wide variety of simple and complex deep learning networks.

[Play Framework](#) is a web framework combines productivity and performance making it easy to build scalable web applications with Java and Scala.

[Dotty](#) is a research compiler that will become Scala 3.

[AWS Scala](#) is a tool that enables Scala developers to easily work with Amazon Web Services in the Scala way.

[Scala.js](#) is a compiler that converts Scala to JavaScript.

[Polynote](#) is an experimental polyglot notebook environment. Currently, it supports Scala and Python (with or without Spark), SQL, and Vega.

[Scala Native](#) is an optimizing ahead-of-time compiler and lightweight managed runtime designed specifically for Scala.

[Gitbucket](#) is a Git platform powered by Scala with easy installation, high extensibility & GitHub API compatibility.

[Finagle](#) is a fault tolerant, protocol-agnostic RPC system

[Gatling](#) is a load test tool. It officially supports HTTP, WebSocket, Server-Sent-Events and JMS.

[Scalatra](#) is a tiny Scala high-performance, async web framework, inspired by [Sinatra](#).

27. Groovy

[Back to the Top](#)



Groovy Guide

A guide covering the Groovy programming language including the applications and tools that will make you a better and more efficient Groovy developer.

Resources

[Apache Groovy](#) is a powerful, optionally typed and dynamic language, with static-typing and static compilation capabilities, for the Java platform aimed at improving developer productivity thanks to a concise, familiar and easy to learn syntax.

[Introducing Groovy from Oracle](#)

[Learning Groovy](#)

[The Complete Apache Groovy Developer Course on Udemy](#)

[Groovy Fundamentals from Pluralsight](#)

[Getting Started with Groovy from Pluralsight](#)

[Fluently Groovy tutorial from IBM](#)

[Groovy language Style guide](#)

Tools

[IntelliJ IDEA](#) is an IDE for Java, but it also understands and provides intelligent coding assistance for a large variety of other languages such as Kotlin, SQL, JPQL, HTML, JavaScript, etc., even if the language

expression is injected into a String literal in your Java code.

[Gradle](#) is a build automation tool that supports multi-language development. If you're building, testing, publishing, and deploying software on any platform, Gradle offers a flexible model that can support the entire development lifecycle from compiling and packaging code to publishing web sites. Gradle is designed to support build automation across multiple languages and platforms including Java, Scala, Android, C/C++, Swift, and Groovy, and is closely integrated with development tools and continuous integration servers including Eclipse, IntelliJ, and Jenkins.

[Groovy Language Server](#) is a [language server](#) protocol for [Groovy](#).

[Apache Camel](#) is an Open Source integration framework that supports Groovy along with other Scripting Languages to allow an Expression or Predicate to be used in the DSL or [Xml Configuration](#).

[Gradle Kotlin DSL converter](#) is a tool that converts from Groovy to Kotlin DSL for Gradle in Android Studio.

[Rest-assured](#) is a Java DSL for easy testing of REST services.

[Spock](#) is a BDD-style developer testing and specification framework for Java and Groovy applications.

[Micronaut](#) is a modern, JVM-based, full stack Java framework designed for building modular, easily testable JVM applications with support for Java, Kotlin and the Groovy language.

[Nextflow](#) is a bioinformatics workflow manager that enables the development of portable and reproducible workflows. It supports deploying workflows on a variety of execution platforms including local, HPC schedulers, AWS Batch, Google Cloud Life Sciences, and Kubernetes. Additionally, it provides support for manage your workflow dependencies through built-in support for Conda, Docker, Singularity, and Modules.

[Dex : The Data Explorer](#) is a data visualization tool written in Java/Groovy/JavaFX capable of powerful ETL and publishing web visualizations.

[Groovy-Eclipse](#) is a project that provides Eclipse and Maven tooling support for the Apache Groovy programming language.

28. Clojure

[Back to the Top](#)



Clojure Guide

A guide covering the Clojure programming language including the applications and tools that will make you a better and more efficient Clojure developer.

Resources

[Clojure](#) is a dynamic, general-purpose programming language, combining the approachability and interactive development of a scripting language with an efficient and robust infrastructure for multithreaded programming.

[Clojure Community Resources](#)

[Clojure Training](#)

[Learning Clojure on Udemy](#)

[Clojure Fundamentals on Pluralsight](#)

[Web Development in Clojure](#)

[Clojure Online Courses on LinkedIn Learning](#)

[Clojure Workshop on Packt](#)

[Clojure Style Guide](#)

Tools

[clojureVSCode](#) is an extension that provides Clojure and ClojureScript support for Visual Studio Code.

[Reagent](#) is a minimalistic [ClojureScript](#) interface to [React](#).

[Riemann](#) is a network event stream processing system for monitoring distributed systems, in Clojure.

[Datascript](#) is an immutable in-memory database and Datalog query engine in Clojure, ClojureScript, and JavaScript.

[Compojure](#) is a small routing library for [Ring](#) that allows web applications to be composed of small, independent parts.

[Ring](#) is a Clojure web applications library inspired by Python's WSGI and Ruby's Rack. By abstracting the details of HTTP into a simple, unified API, Ring allows web applications to be constructed of modular components that can be shared among a variety of applications, web servers, and web frameworks.

[Hiccup](#) is a library for representing HTML in Clojure. It uses vectors to represent elements, and maps to represent an element's attributes.

[Onyx](#) is a distributed, masterless, high performance, fault tolerant data processing system.

[Lumo](#) is a standalone ClojureScript environment that runs on Node.js and the V8 JavaScript engine. It provides out-of-the-box access to the entire Node.js ecosystem, including a ClojureScript REPL.

[Boot](#) is a Clojure build framework and ad-hoc Clojure script evaluator. Boot provides a runtime environment that includes all of the tools needed to build Clojure projects from scripts written in Clojure.

[Arcadia](#) is an integration of the Clojure Programming Language with the Unity 3D game engine.

[Lacinia](#) is a full GraphQL implementation in pure Clojure.

29. WebAssembly

[Back to the Top](#)



WebAssembly Guide

A guide covering WebAssembly including the applications and tools that will make you a better and more efficient developer with WebAssembly.

Resources

[WebAssembly](#) is a new open standard of code that can be run in modern web browsers. It is a low-level assembly-like language with a compact binary format that runs with near-native performance and provides languages such as C/C++, C# and Rust with a compilation target so that they can run on the web. WebAssembly is developed by W3C, Mozilla, Microsoft, Google, and Apple.

[W3C WebAssembly Working Group](#)

[WebAssembly Community Group](#)

[WebAssembly Roadmap](#)

[WebAssembly proposals](#)

[WebAssembly Web API](#)

[Qt for WebAssembly](#)

[Introducing the WebAssembly backend for TensorFlow.js](#)

[Blazor WebAssembly](#)

[Host and deploy ASP.NET Core Blazor WebAssembly](#)

[WebAssembly for Beginners by DigitalOcean](#)

[WebAssembly Courses on LinkedIn Learning](#)

[WebAssembly courses on Udemy](#)

[Gaining A Thorough Knowledge Of Web Assembly course on Pluralsight](#)

[Getting Started Blazor course on Pluralsight](#)

[WebAssembly Training Courses on NobleProg](#)

Tools

[Uno Platform](#) is a Universal Windows Platform Bridge that allows UWP-based code (C# and XAML) to run on iOS, Android, macOS, WebAssembly, Linux and Windows. It provides the full definitions of the UWP [Windows 10 2004 \(19041\)](#), and the implementation of a growing number of parts of the UWP API, such as `Windows.UI.Xaml`, to enable UWP and WinUI applications to run on these platforms.

[WebAssembly Studio](#) is a tool that allows you to Learn, Teach, Work and Play with WebAssembly.

[WABT](#) is a suite of tools for WebAssembly.

[Binaryen](#) is a compiler and toolchain infrastructure library for WebAssembly, written in C++. It aims to make [compiling to WebAssembly](#) easy, fast, and effective.

[Lucet](#) is a native WebAssembly compiler and runtime. It is designed to safely execute untrusted WebAssembly programs inside your application.

[Blazor](#) is a feature of ASP.NET, a popular web development framework that extends the .NET developer platform with tools and libraries for building web apps. Run on WebAssembly or the server Blazor can run your client-side C# code directly in the browser, using WebAssembly.

[Wasmer](#) enables super lightweight containers based on [WebAssembly](#) that can run anywhere such as the Desktop to the Cloud and IoT devices, and also embedded in [any programming language](#).

[AssemblyScript](#) compiles a strict variant of TypeScript (JavaScript with types) to WebAssembly using [Binaryen](#).

[TinyGo](#) is a Go compiler(based on LLVM) intended for use in small places such as microcontrollers, WebAssembly (Wasm), and command-line tools.

[Capstone](#) is a disassembly/disassembler framework: Core (Arm, Arm64, BPF, EVM, M68K, M680X, MOS65xx, Mips, PPC, RISCV, Sparc, SystemZ, TMS320C64x, Web Assembly, X86, X86_64, XCore) + bindings.

[Vecty](#) is a tool that lets you build responsive and dynamic web frontends in Go using WebAssembly, competing with modern web frameworks like React & VueJS.

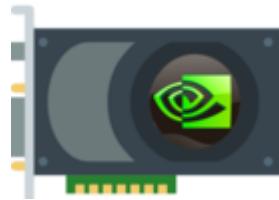
[WebAssembly Micro Runtime \(WAMR\)](#) is a standalone WebAssembly (WASM) runtime with a small footprint.

[Pion WebRTC](#) is a pure Go implementation of [WebRTC API](#).

[vim.wasm](#) is a port of the Vim editor to WebAssembly using [Emscripten](#) and [Binaryen](#).

30. CUDA

[Back to the Top](#)



CUDA Guide

A guide covering CUDA including the applications and tools that will make you a better and more efficient CUDA developer.

Resources

[CUDA](#) is a parallel computing platform and programming model developed by NVIDIA for general computing on graphical processing units (GPUs). With CUDA, developers are able to dramatically speed up computing applications by harnessing the power of GPUs. In GPU-accelerated applications, the sequential part of the workload runs on the CPU, which is optimized for single-threaded. The compute intensive portion of the application runs on thousands of GPU cores in parallel. When using CUDA, developers can program in popular languages such as C, C++, Fortran, Python and MATLAB.

[CUDA Toolkit Documentation](#)

[CUDA Quick Start Guide](#)

[CUDA on WSL](#)

[CUDA GPU support for TensorFlow](#)

[NVIDIA Deep Learning cuDNN Documentation](#)

[NVIDIA GPU Cloud Documentation](#)

[NVIDIA NGC](#) is a hub for GPU-optimized software for deep learning, machine learning, and high-performance computing (HPC) workloads.

[NVIDIA NGC Containers](#) is a registry that provides researchers, data scientists, and developers with simple access to a comprehensive catalog of GPU-accelerated software for AI, machine learning and HPC. These containers take full advantage of NVIDIA GPUs on-premises and in the cloud.

Tools

[CUDA Toolkit](#) is a collection of tools & libraries that provide a development environment for creating high performance GPU-accelerated applications. The CUDA Toolkit allows you can develop, optimize, and deploy your applications on GPU-accelerated embedded systems, desktop workstations, enterprise data centers, cloud-based platforms and HPC supercomputers. The toolkit includes GPU-accelerated libraries, debugging and optimization tools, a C/C++ compiler, and a runtime library to build and deploy your application on major architectures including x86, Arm and POWER.

[NVIDIA cuDNN](#) is a GPU-accelerated library of primitives for [deep neural networks](#). cuDNN provides highly tuned implementations for standard routines such as forward and backward convolution, pooling,

normalization, and activation layers. cuDNN accelerates widely used deep learning frameworks, including [Caffe2](#), [Chainer](#), [Keras](#), [MATLAB](#), [MxNet](#), [PyTorch](#), and [TensorFlow](#).

[CUDA-X HPC](#) is a collection of libraries, tools, compilers and APIs that help developers solve the world's most challenging problems. CUDA-X HPC includes highly tuned kernels essential for high-performance computing (HPC).

[NVIDIA Container Toolkit](#) is a collection of tools & libraries that allows users to build and run GPU accelerated Docker containers. The toolkit includes a container runtime [library](#) and utilities to automatically configure containers to leverage NVIDIA GPUs.

[Minkowski Engine](#) is an auto-differentiation library for sparse tensors. It supports all standard neural network layers such as convolution, pooling, unpooling, and broadcasting operations for sparse tensors.

[CUTLASS](#) is a collection of CUDA C++ template abstractions for implementing high-performance matrix-multiplication (GEMM) at all levels and scales within CUDA. It incorporates strategies for hierarchical decomposition and data movement similar to those used to implement cuBLAS.

[CUB](#) is a cooperative primitives for CUDA C++ kernel authors.

[Tensorman](#) is a utility for easy management of Tensorflow containers by developed by [System76](#). Tensorman allows Tensorflow to operate in an isolated environment that is contained from the rest of the system. This virtual environment can operate independent of the base system, allowing you to use any version of Tensorflow on any version of a Linux distribution that supports the Docker runtime.

[Numba](#) is an open source, NumPy-aware optimizing compiler for Python sponsored by Anaconda, Inc. It uses the LLVM compiler project to generate machine code from Python syntax. Numba can compile a large subset of numerically-focused Python, including many NumPy functions. Additionally, Numba has support for automatic parallelization of loops, generation of GPU-accelerated code, and creation of ufuncs and C callbacks.

[Chainer](#) is a Python-based deep learning framework aiming at flexibility. It provides automatic differentiation APIs based on the define-by-run approach (dynamic computational graphs) as well as object-oriented high-level APIs to build and train neural networks. It also supports CUDA/cuDNN using [CuPy](#) for high performance training and inference.

[CuPy](#) is an implementation of NumPy-compatible multi-dimensional array on CUDA. CuPy consists of the core multi-dimensional array class, `cupy.ndarray`, and many functions on it. It supports a subset of `numpy.ndarray` interface.

[CatBoost](#) is a fast, scalable, high performance [Gradient Boosting](#) on Decision Trees library, used for ranking, classification, regression and other machine learning tasks for Python, R, Java, C++. Supports computation on CPU and GPU.

[cuDF](#) is a GPU DataFrame library for loading, joining, aggregating, filtering, and otherwise manipulating data. cuDF provides a pandas-like API that will be familiar to data engineers & data scientists, so they can use it to easily accelerate their workflows without going into the details of CUDA programming.

[cuML](#) is a suite of libraries that implement machine learning algorithms and mathematical primitives functions that share compatible APIs with other RAPIDS projects. cuML enables data scientists,

researchers, and software engineers to run traditional tabular ML tasks on GPUs without going into the details of CUDA programming. In most cases, cuML's Python API matches the API from scikit-learn.

[ArrayFire](#) is a general-purpose library that simplifies the process of developing software that targets parallel and massively-parallel architectures including CPUs, GPUs, and other hardware acceleration devices.

[Thrust](#) is a C++ parallel programming library which resembles the C++ Standard Library. Thrust's high-level interface greatly enhances programmer productivity while enabling performance portability between GPUs and multicore CPUs.

[AresDB](#) is a GPU-powered real-time analytics storage and query engine. It features low query latency, high data freshness and highly efficient in-memory and on disk storage management.

[Arraymancer](#) is a tensor (N-dimensional array) project in Nim. The main focus is providing a fast and ergonomic CPU, Cuda and OpenCL ndarray library on which to build a scientific computing ecosystem.

[Kintinuous](#) is a real-time dense visual SLAM system capable of producing high quality globally consistent point and mesh reconstructions over hundreds of metres in real-time with only a low-cost commodity RGB-D sensor.

[GraphVite](#) is a general graph embedding engine, dedicated to high-speed and large-scale embedding learning in various applications.

31. Julia

[Back to the Top](#)



Julia Guide

A guide covering Julia programming language including the applications and tools that will make you a better and more efficient Julia developer.

Julia is a high-level, [high-performance](#) dynamic language for technical computing. Julia programs compile to efficient native code for [multiple platforms](#) via LLVM.

[JuliaHub](#) contains over 4,000 Julia packages for use by the community.

[Julia Observer](#)

[Julia Manual](#)

[JuliaLang Essentials](#)

[Julia Style Guide](#)

[Julia By Example](#)

[JuliaLang Gitter](#)

[DataFrames Tutorial using Jupyter Notebooks](#)

[Julia Academy](#)

[Julia Meetup groups](#)

[Julia on Microsoft Azure](#)

Tools

[JuliaPro](#) is a free and fast way to setup Julia for individual researchers, engineers, scientists, quants, traders, economists, students and others. Julia developers can build better software quicker and easier while benefiting from Julia's unparalleled high performance. It includes 2600+ open source packages or from a curated list of 250+ JuliaPro packages. Curated packages are tested, documented and supported by Julia Computing.

[Juno](#) is a powerful, free IDE based on [Atom](#) for the Julia language.

[Debugger.jl](#) is the Julia debuggin tool.

[Profile \(Stdlib\)](#) is a module provides tools to help developers improve the performance of their code. When used, it takes measurements on running code, and produces output that helps you understand how much time is spent on individual line's.

[Revise.jl](#) allows you to modify code and use the changes without restarting Julia. With Revise, you can be in the middle of a session and then update packages, switch git branches, and/or edit the source code in the editor of your choice; any changes will typically be incorporated into the very next command you issue from the REPL. This can save you the overhead of restarting Julia, loading packages, and waiting for code to JIT-compile.

[JuliaGPU](#) is a Github organization created to unify the many packages for programming GPUs in Julia. With its high-level syntax and flexible compiler, Julia is well positioned to productively program hardware accelerators like GPUs without sacrificing performance.

[IJulia.jl](#) is the Julia kernel for Jupyter.

[AWS.jl](#) is a Julia interface for [Amazon Web Services](#).

[CUDA.jl](#) is a package for the main programming interface for working with NVIDIA CUDA GPUs using Julia. It features a user-friendly array abstraction, a compiler for writing CUDA kernels in Julia, and wrappers for various CUDA libraries.

[XLA.jl](#) is a package for compiling Julia to XLA for [Tensor Processing Unit\(TPU\)](#).

[Nanosoldier.jl](#) is a package for running JuliaCI services on MIT's Nanosoldier cluster.

[Julia For VSCode](#) is a powerful extension for the Julia language.

[JuMP.jl](#) is a domain-specific modeling language for [mathematical optimization](#) embedded in Julia.

[Optim.jl](#) is a univariate and multivariate optimization in Julia.

[RCall.jl](#) is a package that allows you to call R functions from Julia.

[JavaCall.jl](#) is a package that allows you to call Java functions from Julia.

[PyCall.jl](#) is a package that allows you to call Python functions from Julia.

[MXNet.jl](#) is the Apache MXNet Julia package. MXNet.jl brings flexible and efficient GPU computing and state-of-art deep learning to Julia.

[Knet](#) is the [Koç University](#) deep learning framework implemented in Julia by [Deniz Yuret](#) and collaborators. It supports GPU operation and automatic differentiation using dynamic computational graphs for models defined in plain Julia.

[Distributions.jl](#) is a Julia package for probability distributions and associated functions.

[DataFrames.jl](#) is a tool for working with tabular data in Julia.

[Flux.jl](#) is an elegant approach to machine learning. It's a 100% pure-Julia stack, and provides lightweight abstractions on top of Julia's native GPU and AD support.

[IRTools.jl](#) is a simple and flexible IR format, expressive enough to work with both lowered and typed Julia code, as well as external IRs.

[Cassette.jl](#) is a Julia package that provides a mechanism for dynamically injecting code transformation passes into Julia's just-in-time (JIT) compilation cycle, enabling post hoc analysis and modification of "Cassette-unaware" Julia programs without requiring manual source annotation or refactoring of the target code.

32. R

[Back to the Top](#)



R Guide

A guide covering R programming language including the applications and tools that will make you a better and more efficient R developer.

Resources

R is an open source software environment for statistical computing and graphics. It compiles and runs on a wide variety of platforms such as Windows and MacOS.

[An Introduction to R](#)

[Google's R Style Guide](#)

[R developer's guide to Azure](#)

[Running R at Scale on Google Compute Engine](#)

[Running R on AWS](#)

[RStudio Server Pro for AWS](#)

[Learn R by Codecademy](#)

[Learn R Programming with Online Courses and Lessons by edX](#)

[R Language Courses by Coursera](#)

[Learn R For Data Science by Udacity](#)

Tools

[RStudio](#) is an integrated development environment for R and Python, with a console, syntax-highlighting editor that supports direct code execution, and tools for plotting, history, debugging and workspace management.

[Shiny](#) is a newer package from RStudio that makes it incredibly easy to build interactive web applications with R.

[Rmarkdown](#) is a package helps you create dynamic analysis documents that combine code, rendered output (such as figures), and prose.

[Rplugin](#) is R Language supported plugin for the IntelliJ IDE.

[Plotly](#) is an R package for creating interactive web graphics via the open source JavaScript graphing library [plotly.js](#).

[Metaflow](#) is a Python/R library that helps scientists and engineers build and manage real-life data science projects. Metaflow was originally developed at Netflix to boost productivity of data scientists who work on a wide variety of projects from classical statistics to state-of-the-art deep learning.

[Prophet](#) is a procedure for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. It works best with time series that have strong seasonal effects and several seasons of historical data.

[LightGBM](#) is a gradient boosting framework that uses tree based learning algorithms, used for ranking, classification and many other machine learning tasks.

[Dash](#) is a Python framework for building analytical web applications in Python, R, Julia, and Jupyter.

[MLR](#) is Machine Learning in R.

[ML workspace](#) is an all-in-one web-based IDE specialized for machine learning and data science. It is simple to deploy and gets you started within minutes to productively built ML solutions on your own machines. ML workspace is the ultimate tool for developers preloaded with a variety of popular data science libraries

(Tensorflow, PyTorch, Keras, and MXnet) and dev tools (Jupyter, VS Code, and Tensorboard) perfectly configured, optimized, and integrated.

[CatBoost](#) is a fast, scalable, high performance Gradient Boosting on Decision Trees library, used for ranking, classification, regression and other machine learning tasks for Python, R, Java, C++. Supports computation on CPU and GPU.

[Plumber](#) is a tool that allows you to create a web API by merely decorating your existing R source code with special comments.

[Drake](#) is an R-focused pipeline toolkit for reproducibility and high-performance computing.

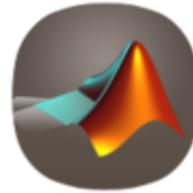
[DiagrammeR](#) is a package you can create, modify, analyze, and visualize network graph diagrams. The output can be incorporated into R Markdown documents, integrated with Shiny web apps, converted to other graph formats, or exported as image files.

[Knitr](#) is a general-purpose literate programming engine in R, with lightweight API's designed to give users full control of the output without heavy coding work.

[Broom](#) is a tool that converts statistical analysis objects from R into tidy format.

33. MATLAB

[Back to the Top](#)



MATLAB Guide

A guide covering MATLAB including the applications and tools that will make you a better and more efficient MATLAB developer.

Resources

[MATLAB](#) is a programming language that does numerical computing such as expressing matrix and array mathematics directly.

[MATLAB Documentation](#)

[Getting Started with MATLAB](#)

[MATLAB and Simulink Training from MATLAB Academy](#)

[MathWorks Certification Program](#)

[MATLAB Online Courses from Udemy](#)

[MATLAB Online Courses from Coursera](#)

[MATLAB Online Courses from edX](#)

[Building a MATLAB GUI](#)

[MATLAB Style Guidelines 2.0](#)

[Setting Up Git Source Control with MATLAB & Simulink](#)

[Pull, Push and Fetch Files with Git with MATLAB & Simulink](#)

[Create New Repository with MATLAB & Simulink](#)

[PRMLT](#) is Matlab code for machine learning algorithms in the PRML book.

Tools

[MATLAB Online](#) allows to users to utilize MATLAB and Simulink through a web browser such as Google Chrome.

[Simulink](#) is a block diagram environment for Model-Based Design. It supports simulation, automatic code generation, and continuous testing of embedded systems.

[MATLAB Schemer](#) is a MATLAB package makes it easy to change the color scheme (theme) of the MATLAB display and GUI.

[LRSLibrary](#) is a Low-Rank and Sparse Tools for Background Modeling and Subtraction in Videos. The library was designed for moving object detection in videos, but it can be also used for other computer vision and machine learning problems.

[Robotics Toolbox for MATLAB](#) provides a toolbox that brings robotics specific functionality(designing, simulating, and testing manipulators, mobile robots, and humanoid robots) to MATLAB, exploiting the native capabilities of MATLAB (linear algebra, portability, graphics). The toolbox also supports mobile robots with functions for robot motion models (bicycle), path planning algorithms (bug, distance transform, D*, PRM), kinodynamic planning (lattice, RRT), localization (EKF, particle filter), map building (EKF) and simultaneous localization and mapping (EKF), and a Simulink model a of non-holonomic vehicle. The Toolbox also including a detailed Simulink model for a quadrotor flying robot.

[SEA-MAT](#) is a collaborative effort to organize and distribute Matlab tools for the Oceanographic Community.

[Gramm](#) is a complete data visualization toolbox for Matlab. It provides an easy to use and high-level interface to produce publication-quality plots of complex data with varied statistical visualizations. Gramm is inspired by R's ggplot2 library.

[hctsa](#) is a software package for running highly comparative time-series analysis using Matlab.

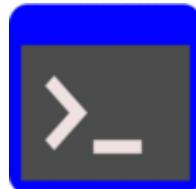
[Plotly](#) is a Graphing Library for MATLAB.

[YALMIP](#) is a MATLAB toolbox for optimization modeling.

[GNU Octave](#) is a high-level interpreted language, primarily intended for numerical computations. It provides capabilities for the numerical solution of linear and nonlinear problems, and for performing other numerical experiments. It also provides extensive graphics capabilities for data visualization and manipulation.

34. Bash/Shell/PowerShell

[Back to the Top](#)



Bash/Shell/PowerShell Guide

A guide covering Bash/Shell/PowerShell scripting languages including the applications and tools that will make you a better and more efficient Bash/Shell/PowerShell developer.

Resources

[Unix Shell](#) is a command-line interpreter/shell that provides a user CLI for Unix-like operating systems such as MacOS.

[GNU Bash Manual](#)

[Introduction to Bash Shell Scripting by Coursera](#)

[Bash: Shell Script Basics by Pluralsight](#)

[Bash/Shell by Codecademy](#)

[Korn shell scripting](#)

[C Shell scripting](#)

[Getting Started with PowerShell](#)

[PowerShell in Azure Cloud Shell](#)

[Azure Functions using PowerShell](#)

[Azure Automation runbooks](#)

[Using Visual Studio Code For PowerShell Development](#)

[Integrated Terminal in Visual Studio Code](#)

[AWS Tools for Windows PowerShell](#)

[PowerShell Best Practices and Style Guide](#)

AWS Command Line Interface and aws-shell Sample for AWS Cloud9

[Configuring Cloud Shell on Google Cloud](#)

[Google's Shell Style Guide](#)

Tools

[Bash](#) is the GNU Project's shell(Bourne Again SHell), which is an sh-compatible shell that integrates together useful features from the Korn shell (ksh) and the C shell (csh).

[PowerShell Core](#) is a cross-platform (Windows, Linux, and macOS) automation and configuration tool/framework that works well with your existing tools and is optimized for dealing with structured data (JSON, CSV, XML, etc.), REST APIs, and object models. It also includes a command-line shell, an associated scripting language and a framework for processing cmdlets.

[Azure PowerShell](#) is a set of cmdlets for managing Microsoft Azure resources directly from the PowerShell command line.

[Windows Subsystem for Linux \(WSL\)](#) is a compatibility layer developed by Microsoft for running Linux binary executables in a Executable/Linkable Format natively on Windows 10 and Windows Server.

[AWS Shell](#) is a command-line shell program that provides convenience and productivity features to help both new and advanced users of the AWS Command Line Interface.

[Google Cloud Shell](#) is a free admin machine with browser-based command-line access for managing your infrastructure and applications on Google Cloud Platform.

[Zsh](#) is a shell designed for interactive use and also a powerful scripting language.

[Oh My Zsh](#) is an awesome community-driven (with 1700+ contributors) framework for managing your zsh configuration. Includes 200+ optional plugins (rails, git, hub, capistrano, brew, ant, php, python, etc), over 140 themes to spice up your morning, and an auto-update tool so that makes it easy to keep up with the latest updates from the community.

[fish shell](#) is a smart and user-friendly command line shell for macOS, Linux, and Windows. fish includes features like syntax highlighting, autosuggest-as-you-type, and fancy tab completions that just work, with no configuration required.

[Dash \(Debian Almquist shell\)](#) is a modern [POSIX-compliant](#) implementation of /bin/sh that aims to be as small as possible.

[VS Code Bash Debug](#) is a bash debugger GUI frontend based on awesome bashdb scripts (bashdb now included in package).

[VS Code Bash IDE](#) is a Visual Studio Code extension utilizing the [bash language server](#), that is based on [Tree Sitter](#) and its [grammar for Bash](#) and supports [explainshell](#) integration.

35. C#

[Back to the Top](#)



C# Guide

A guide covering C# including the applications and tools that will make you a better and more efficient C# developer.

Resources

C# is a modern and object-oriented programming language developed by Microsoft to write any application using the C# programming language on the .NET platform.

[Taking your first steps with C#](#)

[Learning C#](#)

[C# development with Visual Studio](#)

[C# programming with Visual Studio Code](#)

[Working with data in C#](#)

[C# Tutorial by W3Schools](#)

[Windows Forms for .NET 5 and .NET Core 3.1](#)

[Xamarin documentation](#)

[Advanced Topics in C# by Udemy](#)

[The complete C# tutorial](#)

[Unity C# Survival Guide](#)

[RabbitMQ .NET/C# Client API](#)

Tools

[Mono](#) is a software platform designed to allow developers to easily create cross platform applications. It is an open source implementation of Microsoft's .NET Framework based on the ECMA standards for C# and the Common Language Runtime.

[Visual Studio](#) is an integrated development environment (IDE) from Microsoft; which is a feature-rich application that can be used for many aspects of software development. Visual Studio makes it easy to edit, debug, build, and publish your app. By using Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, and Windows Store.

[MSBuild](#) is the build platform for .NET and Visual Studio. MSBuild, provides an XML schema for a project file that controls how the build platform processes and builds software. Visual Studio uses MSBuild to perform

team builds through Azure DevOps Server, but MSBuild can run without Visual Studio.

[Roslyn](#) is a .NET compiler developed by Microsoft that provides C# and Visual Basic languages with rich code analysis APIs.

[Bot Framework](#) is a framework developed by Microsoft that provides the most comprehensive experience for building conversation applications. Developers can model and build sophisticated conversation using their favorite programming languages including C#, JS, Python and Java or using Bot Framework Composer, an open-source, visual authoring canvas for developers and multi-disciplinary teams to design and build conversational experiences with Language.

[Uno Platform](#) is a Universal Windows Platform Bridge that allows UWP-based code (C# and XAML) to run on iOS, Android, macOS, WebAssembly, Linux and Windows 7. It provides the full definitions of the UWP [Windows 10 2004 \(19041\)](#), and the implementation of a growing number of parts of the UWP API, such as Windows.UI.Xaml, to enable UWP and WinUI applications to run on these platforms.

[Rider](#) is a fast and powerful, cross-platform .NET IDE developed by JetBrains to develop .NET, ASP.NET, .NET Core, Xamarin; or Unity applications for Windows, Mac, Linux.

[Resharper](#) is a [Visual Studio](#) Extension for .NET Developers that has On-the-fly code quality analysis for C#, VB.NET, XAML, ASP.NET, ASP.NET MVC, JavaScript, TypeScript, CSS, HTML, and XML. Letting you know right away if your code needs to be improved.

[dotPeek](#) is a tool developed by JetBrains based on ReSharper's bundled decompiler. It can reliably decompile any .NET assembly into equivalent C# or CIL code.

[dotTrace](#) is an .NET performance Profiler developed by Jet Brains. It helps users locate performance bottlenecks in a variety of .NET applications: desktop applications, .NET Core, ASP.NET, ASP.NET Core applications hosted on IIS or IIS Express web servers, Silverlight, WCF services, Windows services, Universal Windows Platform applications, and unit tests.

[dotMemory](#) is an .NET memory Profiler developed by Jet Brains. It allows the user to analyze memory usage in a variety of .NET and .NET Core applications: desktop applications, Windows services, ASP.NET web applications, IIS, IIS Express, arbitrary .NET processes, and more.

[dotCover](#) is an .NET unit test runner and code coverage tool developed by Jet Brains. It helps the user figure out on-the-fly which unit tests are affected by your latest code changes, and automatically re-runs the affected tests for you. The continuous testing mode can be switched on for any unit test session.

[Json.NET](#) is a popular high-performance JSON framework for .NET.

[Quasar](#) is a fast and light-weight remote administration tool coded in C#. The usage ranges from user support through day-to-day administrative work to employee monitoring. Providing high stability and an easy-to-use user interface, Quasar is the perfect remote administration solution for you.

[CodeMaid](#) is an open source Visual Studio extension to cleanup and simplify our C#, C++, F#, VB, PHP, PowerShell, JSON, XAML, XML, ASP, HTML, CSS, LESS, SCSS, JavaScript and TypeScript coding.

[.NET Fiddle](#) is an advanced online compiler for C# that allows you to create, run and share your code online.

[Octopus Deploy](#) is a single place for your team to manage releases, automate deployments, and automate the runbooks that keeps your software operating.

[Appveyor](#) is a cloud-based continuous integration system that integrates natively with your source control and allows CI configuration files to live alongside your projects.

[AppHarbor](#) is a .NET Platform-as-a-Service that let's developers deploy and scale any standard .NET application to the cloud.

[ANTLR \(ANother Tool for Language Recognition\)](#) is a powerful parser generator for reading, processing, executing, or translating structured text or binary files. It's widely used to build languages, tools, and frameworks. From a grammar, ANTLR generates a parser that can build parse trees and also generates a listener interface that makes it easy to respond to the recognition of phrases of interest.

[AutoRest](#) is a tool generates client libraries for accessing RESTful web services using the [OpenAPI Specification](#) format. It Supports C#, PowerShell, Go, Java, Node.js, TypeScript, Python, Ruby.

[Markdig](#) is a fast, powerful, [CommonMark](#) compliant, extensible Markdown processor for .NET.

36. F#

[Back to the Top](#)



F# Guide

A guide covering F# including the applications and tools that will make you a better and more efficient F# developer.

Resources

[F#](#) is a mature, open source, cross-platform, functional-first programming language. It empowers users and organizations to tackle complex computing problems with simple, maintainable and robust code.

[What is F#](#)

[F#, the Functional programming for .NET](#)

[Cloud Programming with F#](#)

[F# style guide](#)

[F# Programming Wikibook](#)

[F# Developer Network \(FSDN\)](#)

Learning F# from The F# Software Foundation

F# Programming groups | Meetup

Tools

[Python.NET](#) is a package that gives Python programmers nearly seamless integration with the .NET Common Language Runtime (CLR) and provides a powerful application scripting tool for .NET developers. It allows Python code to interact with the CLR, and may also be used to embed Python into a .NET application.

[Fable](#) is an F# to JavaScript compiler powered by [Babel](#), designed to make F# a first-class citizen of the JavaScript ecosystem.

[.NET for Apache Spark](#) provides high performance APIs for using Apache Spark from C# and F#. With these .NET APIs, you can access the most popular Dataframe and SparkSQL aspects of Apache Spark, for working with structured data, and Spark Structured Streaming, for working with streaming data. .NET for Apache Spark runs on Windows, Linux, and macOS using .NET Core, or Windows using .NET Framework. It also runs on all major cloud providers including [Azure HDInsight Spark](#), [Amazon EMR Spark](#), [AWS](#) & [Azure Databricks](#).

[Giraffe](#) is a functional ASP.NET Core micro web framework for building rich web applications.

[Suave](#) is a simple web development F# library providing a lightweight web server and a set of combinators to manipulate route flow and task composition. Suave supports Websocket, HTTPS, multiple TCP/IP bindings, Basic Access Authentication, Keep-Alive.

[FsCheck](#) is a tool for testing .NET programs automatically. The programmer provides a specification of the program, in the form of properties which functions, methods or objects should satisfy, and FsCheck then tests that the properties hold in a large number of randomly generated cases.

[Elmish](#) implements core abstractions for F# apps that can be used to build applications following the "model view update" style of architecture made famous by Elm. The library however does not model any "view" and is intended for use in conjunction with a DOM/renderer, like React/ReactNative or VirtualDOM.

[Saturn](#) is a web development framework written in F# which implements the server-side MVC pattern. Many of its components and concepts will seem familiar to anyone with experience in other web frameworks like Ruby on Rails or Python's Django.

[Pulumi](#) is Infrastructure as Code SDK is the easiest way to create and deploy cloud software that use containers, serverless functions, hosted services, and infrastructure, on any cloud. Simply write code in your favorite language and Pulumi will automatically provisions and manages your AWS, Azure, Google Cloud Platform, and/or Kubernetes resources, using an infrastructure-as-code approach.

[FsPickler](#) is a serialization library that facilitates the distribution of .NET objects. The implementation focuses on performance and completeness in supported types, including F# types. It supports multiple, pluggable serialization formats including Binary, Xml, JSON and BSON. The library is based on the functional programming concept of [pickler combinators](#) which has been adapted to accommodate the object oriented nature of the .NET framework.

[Paket](#) is a dependency manager for .NET with support for NuGet packages and git repositories.

[Fantomas](#) is a F# source code formatter.

[SQLProvider](#) is a general F# SQL database erasing type provider, supporting LINQ queries, schema exploration, individuals, and CRUD operations.

[FsUnit](#) makes unit-testing with F# more enjoyable. It adds a special syntax to your favorite .NET testing framework.

37. .NET/.NET Core

[Back to the Top](#)



A guide covering the .NET framework including the applications and tools that will make you a better and more efficient .NET developer.

Resources

[.NET](#) is a developer platform with tools and libraries for building any type of app, including web, mobile, desktop, games, IoT, cloud, and microservices.

[.NET documentation](#)

[Getting started with .NET](#)

[.NET Application Architecture Guide](#)

[Intro .NET Guide by JetBrains](#)

[C# documentation](#) write any application using the C# programming language on the .NET platform.

Tools

[.NET Core](#) is a cross-platform .NET implementation for websites, servers, and console apps on Windows, Linux, and macOS. The .NET Framework supports websites, services, desktop apps, and more on Windows. Xamarin/Mono is a .NET implementation for running apps on all the major mobile operating systems.

[.NET runtime](#) is a collection of libraries and shared host (dotnet) installers for all supported platforms, as well as the sources to .NET runtime and libraries.

[ASP.NET Core](#) is a cross-platform .NET framework for building modern cloud-based web applications on Windows, Mac, or Linux.

[Mono](#) is a software platform designed to allow developers to easily create cross platform applications. It is an open source implementation of Microsoft's .NET Framework based on the ECMA standards for C# and the Common Language Runtime.

[Visual Studio](#) is an integrated development environment (IDE) from Microsoft; which is a feature-rich application that can be used for many aspects of software development. Visual Studio makes it easy to edit, debug, build, and publish your app. By using Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, and Windows Store.

[Rider](#) is a fast and powerful, cross-platform .NET IDE developed by JetBrains to develop .NET, ASP.NET, .NET Core, Xamarin; or Unity applications for Windows, Mac, Linux.

[Resharper](#) is a [Visual Studio](#) Extension for .NET Developers that has On-the-fly code quality analysis for C#, VB.NET, XAML, ASP.NET, ASP.NET MVC, JavaScript, TypeScript, CSS, HTML, and XML. Letting you know right away if your code needs to be improved.

[dotTrace](#) is an .NET performance Profiler developed by Jet Brains. It helps users locate performance bottlenecks in a variety of .NET applications: desktop applications, .NET Core, ASP.NET, ASP.NET Core applications hosted on IIS or IIS Express web servers, Silverlight, WCF services, Windows services, Universal Windows Platform applications, and unit tests.

[dotMemory](#) is an .NET memory Profiler developed by Jet Brains. It allows the user to analyze memory usage in a variety of .NET and .NET Core applications: desktop applications, Windows services, ASP.NET web applications, IIS, IIS Express, arbitrary .NET processes, and more.

[dotCover](#) is an .NET unit test runner and code coverage tool developed by Jet Brains. It helps the user figure out on-the-fly which unit tests are affected by your latest code changes, and automatically re-runs the affected tests for you. The continuous testing mode can be switched on for any unit test session.

[Avalonia](#) is a cross-platform XAML-based UI framework providing a flexible styling system and supporting a wide range of Operating Systems such as Windows via .NET Framework and .NET Core, Linux via Xorg, macOS.

[Polly](#) is a .NET resilience and transient-fault-handling library that allows developers to express policies such as Retry, Circuit Breaker, Timeout, Bulkhead Isolation, and Fallback in a fluent and thread-safe manner.

[IdentityServer](#) is a free, open source [OpenID Connect](#) and [OAuth 2.0](#) framework for ASP.NET Core. IdentityServer4 incorporates all the protocol implementations and extensibility points needed to integrate token-based authentication, single-sign-on and API access control in your applications.

[ILSpy](#) is the open-source .NET assembly browser and decompiler.

[Hangfire](#) is an easy way to perform background job processing in your .NET and .NET Core applications with no Windows Service or Task Scheduler required.

[React Native Windows](#) is a framework for building native Windows apps with React. [React Native](#) is a framework developed by Facebook that enables you to build world-class application experiences on native platforms using a consistent developer experience based on JavaScript and React.

[ReactiveUI](#) is a composable, cross-platform model-view-viewmodel framework for all .NET platforms that is inspired by functional reactive programming, which is a paradigm that allows you to abstract mutable state

away from your user interfaces and express the idea around a feature in one readable place and improve the testability of your application.

[Refit](#) is the automatic type-safe REST library for .NET Core, Xamarin and .NET. It's heavily inspired by Square's Retrofit library, Refit turns your REST API into a live interface.

[MAUI](#) is the .NET Multi-platform App UI, a framework for building native device applications spanning mobile, tablet, and desktop.

[Quasar](#) is a fast and light-weight remote administration tool coded in C#. The usage ranges from user support through day-to-day administrative work to employee monitoring. Providing high stability and an easy-to-use user interface, Quasar is the perfect remote administration solution for you.

38. Dart

[Back to the Top](#)



Dart Guide

A guide covering Dart programming language including the applications and tools that will make you a better and more efficient Dart developer.

Resources

[Dart](#) is an open-source, scalable programming language, with robust libraries and runtimes, for building web, server, and mobile apps using the [Flutter](#) framework.

[Dart API reference](#)

[Dart core libraries](#)

[Dart tutorials](#)

[Getting Started with Dart SDK](#)

[Creating a new Dart application with IntelliJ IDEA](#)

[Effective Dart Style Guide](#)

[Learning Dart](#)

[Build Native Mobile Apps with Flutter on Udacity](#)

[Dart for Beginners on Udemy](#)

[Dart Classes on Skilshare](#)

[Fuchsia Project](#)

[Getting Started with Fuchsia](#)

[Fuchsia Reference](#)

[Contributing to Fuchsia](#)

Tools

[Android Studio](#) is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Available on Windows, macOS, Linux, Chrome OS.

[IntelliJ IDEA](#) is an IDE for Java, but it also understands and provides intelligent coding assistance for a large variety of other languages such as Kotlin, SQL, JPQL, HTML, JavaScript, etc., even if the language expression is injected into a String literal in your Java code.

[Dart DevTools](#) is a suite of debugging and performance tools for Dart and Flutter. These tools are distributed in IDEs, the flutter tool, the webdev tool, and the [devtools package](#).

[DartPad](#) is an online editor for Dart.

[Flutter SDK](#) is Google's UI toolkit for crafting beautiful, natively compiled applications for mobile, web, and desktop from a single codebase using Dart.

[Pub](#) is a tool that has commands for managing packages and for deploying packages and command-line apps.

[Analysis server](#) is designed to provide on-going analysis of one or more code bases as those code bases are changing.

[Dart VM](#) is a tool to run Dart command-line apps such as server-side scripts, programs, and servers.

[dart2js](#) is a tool to compile Dart code to deployable JavaScript. Another Dart-to-JavaScript compiler, [dartdevc](#), is for development use only. The [webdev](#) build command uses dart2js by default.

[dartdevc](#) is the Dart development compiler that lets you run and debug your Dart web app in the Chrome browser.

[AngularDart](#) is a productive web application framework that powers some of Google's most critical applications. It's built on Dart and used extensively by Google AdWords, AdSense, Fiber, and many more projects.

[RxDart](#) is a package with Reactive Extensions for Dart that adds additional capabilities to Dart [Streams](#) and [StreamControllers](#).

39. Flutter

[Back to the Top](#)



Flutter Guide

A guide covering the Flutter Framework including the applications and tools that will make you a better and more efficient Kotlin developer.

Resources

[Flutter](#) is Google's UI toolkit for crafting beautiful, natively compiled applications for mobile(Andorid and iOS), web, and desktop(Windows, MacOS, Linux, and Google Fuchsia) from a single codebase. Flutter works with existing code, is used by developers and organizations around the world, and is free and open source.

[Flutter Gems](#) is a curated package guide for Flutter which functionally categorizes some of the most useful and popular flutter packages available on pub.dev Flutter Gems A Flutter package landscape guide comprising 1500+ neatly categorized useful and popular packages.

[Dart](#) is an open-source, scalable programming language, with robust libraries and runtimes, for building web, server, and mobile apps using the Flutter framework.

[Flutter documentation](#)

[Style Guide for Flutter](#)

[Creating your first Flutter app](#)

[Build and release an Android app using Flutter](#)

[Flutter Tools & techniques](#)

[Dart and Flutter: The Complete Developer's Guide on Udemy](#)

[Creating an Interactive Story with Flutter on Coursera](#)

[Flutter for Beginners course on Pluralsight](#)

[Flutter Online Training Courses on LinkedIn Learning](#)

[The Complete Flutter App Development Bootcamp with Dart by App Brewery](#)

[Adding Firebase to your Flutter app](#)

[Using Firebase and Firestore with Flutter](#)

[Fuchsia Project](#)

[Getting Started with Fuchsia](#)

[Fuchsia Reference](#)

Contributing to Fuchsia

Tools

[Firebase](#) is a Backend-as-a-Service (BaaS) app development platform that provides hosted backend services such as a realtime database, cloud storage, authentication, crash reporting, machine learning, remote configuration, and hosting for your static files.

[FlutterFire](#) is a set of [Flutter plugins](#) that enable Flutter apps to use [Firebase](#) services. You can follow an example that shows how to use these plugins in the [Firebase for Flutter](#) codelab.

[FlutterBoost](#) is a Flutter plugin which enables hybrid integration of Flutter for your existing native apps with minimum efforts.

[Go-flutter](#) is a package that brings Flutter to the desktop. project implements the [Flutter's Embedding API](#) using a single code base that runs on Windows, macOS, and Linux. For rendering, [GLFW](#) fits the job because it provides the right abstractions over the OpenGL's Buffer/Mouse/Keyboard for each platform.

[Appwrite](#) is a secure end-to-end backend server for Web, Mobile, and Flutter developers that is packaged as a set of Docker containers for easy deployment.

[Fluro](#) is a Flutter routing library that adds flexible routing options like wildcards, named parameters and clear route definitions.

[Flame](#) is a minimalistic Flutter game engine.

40. Firebase

[Back to the Top](#)



Firebase Guide

A guide covering Firebase including the applications, libraries and tools that will make you a better and more efficient developer with Firebase.

Resources

[Firebase](#) is a Backend-as-a-Service (BaaS) app development platform that provides hosted backend services such as a realtime database, cloud storage, authentication, crash reporting, machine learning, remote configuration, and hosting for your static files.

[Firebase GitHub](#)

[Add Firebase to your Flutter app](#)

[Using Firebase and Firestore with Flutter](#)

[Using Firebase with the Ionic Framework](#)

[Using Google Analytics with Firebase](#)

[Firebase Cloud Messaging \(FCM\)](#)

[Google Developers Training](#)

[Google Developers Certification](#)

[Firebase Courses on Udemy](#)

[Build a Google Firebase Web Application on Coursera](#)

[Firebase in a Weekend: Android on Udacity](#)

[Firebase Online Training Courses on LinkedIn Learning](#)

[Learning Firebase Fundamentals on Pluralsight](#)

[Fuchsia Project](#)

[Getting Started with Fuchsia](#)

[Fuchsia Reference](#)

[Contributing to Fuchsia](#)

Tools

[Android Studio](#) is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Available on Windows, macOS, Linux, Chrome OS.

[Firebase Analytics](#) is a Google API that provides methods for logging events and setting user properties.

[Firebase ML](#) is an Google API includes all of Firebase's cloud-based ML features.

[ML Kit](#) is a standalone library for on-device ML, which you can use with or without Firebase.

[Firebase iOS](#) is all the Apple platform Firebase SDK source except `FirebaseAnalytics`, `FirebasePerformance`, and `FirebaseML`.

[FlutterFire](#) is a set of [Flutter plugins](#) that enable Flutter apps to use [Firebase](#) services. You can follow an example that shows how to use these plugins in the [Firebase for Flutter](#) codelab.

[AngularFire](#) is the official Angular library for [Firebase](#).

[Vuefire](#) is Firebase bindings for Vue.js & Vuex.

[Reactive Native Firebase](#) is a well-tested feature-rich modular Firebase implementation for React Native. Supports both iOS & Android platforms for all Firebase services.

[React Redux Firebase](#) is Redux bindings for Firebase that includes React Hooks and Higher Order Components.

[FirebaseUI](#) is an open-source library for Android that allows you to quickly connect common UI elements to Firebase APIs.

[Pyrebase](#) is a simple python wrapper for the Firebase API.

[EmberFire](#) is the officially supported adapter for using Firebase with [Ember](#).

[ReactFire](#) is a project that provides Hooks, Context Providers, and Components that make it easy to interact with Firebase.

41. Objective-C

[Back to the Top](#)



A guide covering the Objective-C programming language including the applications and tools that will make you a better and more efficient Objective-C developer.

Resources

[Objective-C](#) was the primary programming language used for writing software for macOS and iOS until [Swift](#) was introduced in 2014. It is a superset of the C programming language and provides object-oriented capabilities and a dynamic runtime.

[Apple Developer Forums](#)

[Google's Objective-C Style Guide](#)

[Objective C Courses on Coursera](#)

[Objective-C online course on Udemy](#)

[Objective-C for Swift Developers course by David Nutter](#)

[Objective-C Essential Training on LinkedIn Learning](#)

[Objective-C for Swift Developers on Udacity](#)

Tools

[Xcode](#) includes everything developers need to create great applications for Mac, iPhone, iPad, Apple TV, and Apple Watch. Xcode provides developers a unified workflow for user interface design, coding, testing,

and debugging.

[AppKit](#) is a graphical user interface toolkit that contains all the objects you need to implement the user interface for a macOS app such as windows, panels, buttons, menus, scrollers, and text fields, and it handles all the details for you as it efficiently draws on the screen, communicates with hardware devices and screen buffers, clears areas of the screen before drawing, and clips views.

[Instruments](#) is a powerful and flexible performance-analysis and testing tool that's part of the Xcode tool set. It's designed to help you profile your iOS, watchOS, tvOS, and macOS apps, processes, and devices in order to better understand and optimize their behavior and performance.

[Cocoapods](#) is a dependency manager for Swift and Objective-C in your Xcode projects by specifying the dependencies for your project in a simple text file. CocoaPods then recursively resolves dependencies between libraries, fetches source code for all dependencies, and creates and maintains an Xcode workspace to build your project.

[AppCode](#) is constantly monitoring the quality of your code. It warns you of errors and smells and suggests quick-fixes to resolve them automatically. AppCode provides lots of code inspections for Objective-C, Swift, C/C++, and a number of code inspections for other supported languages.

[Realm](#) is a mobile database(replaces Core Data & SQLite) that runs directly inside phones, tablets or wearables.

[Infer](#) is a static analysis tool for Java, C++, Objective-C, and C.

[Mantle](#) is a model framework that makes it easy to write a simple model layer for your Cocoa or Cocoa Touch application.

[Quick](#) is a behavior-driven development framework for Swift and Objective-C.

[Aspects](#) is a simple library for aspect oriented programming in Objective-C and Swift.

[Hammerspoon](#) is a tool for powerful automation for macOS that acts as a bridge between the operating system and a Lua scripting engine.

[Nimbus](#) is an iOS framework whose feature set grows only as fast as its documentation.

42. Qt

[Back to the Top](#)



A guide covering the Qt programming language including the applications and tools that will make you a better and more efficient Qt developer.

[Qt](#) is a framework that allows developers to create modern UIs & applications for multiple screens across multiple platforms such as Linux, Windows, macOS, Android, and other embedded systems.

[Qt QML](#)

[Qt Features](#)

[Deploy With Qt](#)

[Qt for Python](#)

[Qt for WebAssembly](#)

[Qt Resource Center](#)

[Qt training by The Qt Company](#)

[Qt for Educational Programs](#)

[Get Qt certified](#)

[QT Framework Courses Online on Udemy](#)

[Learning Qt Quick Fundamentals on Pluralsight](#)

[Qt Creator Manual](#)

Tools

[Qt Creator](#) is a cross-platform, integrated development environment (IDE) for developers to create applications for multiple desktop, embedded, and mobile device platforms.

[Daggy](#) is a server-less remote or local data aggregation and streaming utility.

[GitClient](#) is a multi-platform Git client written with Qt.

[Incredibuild](#) is a plug-and-play solution that empowers Qt users to dramatically reduce their Qt applications build times.

[KDED](#) is a central daemon of KDE workspaces running in the background and performs a number of small tasks that are built-in and others that run on demand.

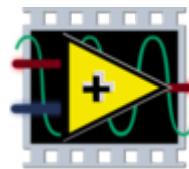
[GammaRay™](#) is a debugger that allows the user to observe behavior and data structures of Qt code inside their program live at runtime.

[Qaterial](#) is a collection of Material Components to build faster UI using Qml.

[Mini-QML](#) is a minimal Qt deployment for Linux, Windows, macOS and WebAssembly.

43. LabVIEW

[Back to the Top](#)



LabVIEW Guide

A guide covering LabVIEW including the applications and tools that will make you a better and more efficient LabVIEW developer.

Resources

[Laboratory Virtual Instrument Engineering Workbench \(LabVIEW\)](#) is a graphical programming approach that helps you visualize every aspect of your application, including hardware configuration, measurement data, and debugging.

[LabVIEW Wiki](#)

[Free Online NI LabVIEW Training for Students and Educators](#)

[National Instruments\(NI\) Badge Program](#)

[Certified LabVIEW Architect \(CLA\)](#)

[Comparing LabVIEW NXG and LabVIEW 2020](#)

[NI Tools Network](#)

[LabVIEW online course on Udemy](#)

[LabVIEW Training Courses on NobleProg](#)

[NI Community](#)

[LabVIEW Open Source Project GitHub](#)

Tools

[LabVIEW NXG](#) is the next generation of LabVIEW that quickly automates your hardware, customizing tests to your specifications, and easily viewing measurement results from anywhere.

[LabVIEW NXG Web Module](#) allows the creation of web-based user interfaces for LabVIEW and LabVIEW NXG applications without the need for web development skills.

[LabVIEW FPGA Module](#) allows the user to develop FPGA VIs on a host computer and compile and implement the code on NI reconfigurable I/O (RIO) hardware.

[LabVIEW Real-Time Module](#) allows the user to create and debug reliable, deterministic applications that run on stand-alone embedded hardware targets.

[Vision Development Module](#) is an add-on that provides hundreds of functions for developing and deploying machine vision applications.

[npTDMS](#) is a cross-platform Python package for reading and writing TDMS files as produced by LabVIEW, and is built on top of the [numpy](#) package.

[Featherweight](#) is an Application Framework for NI LabVIEW™.

[LV-MQTT-Broker](#) is the native LabVIEW MQTT Broker Server.

44. Bootstrap

[Back to the Top](#)



Bootstrap Guide

A guide covering Bootstrap front-end toolkit including the applications and tools that will make you a better and more efficient Bootstrap developer.

Resources

[Bootstrap](#) is the world's most popular front-end open source toolkit, featuring Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful JavaScript plugins.

[BootstrapCDN](#)

[Official Bootstrap Themes](#)

[What is Bootstrap by W3Schools](#)

[Getting Started with Bootstrap](#)

[Getting Started with React-Bootstrap](#)

[Bootswatch](#) is a collection of open source themes for Bootstrap.

[Learn Bootstrap by Codecademy](#)

[Bootstrap Courses by Coursera](#)

[Bootstrap Courses by Udemy](#)

Tools

[Bootstrap Studio](#) is a powerful desktop app for designing and prototyping websites. It comes with a large number of built-in components, which you can easily drag and drop to build responsive web pages.

[Bootstrap Build](#) is a tool that offers many advanced Bootstrap themeing capabilities, such as live preview, variable search, color picker, size adjuster and more.

[LayoutIT!](#) is a interface builder for CSS Grid and Bootstrap that wants to be the kick-off for your front-end developments.

[Tabler](#) is free and open-source HTML Dashboard UI Kit built on Bootstrap.

[BootstrapVue](#) provides one of the most comprehensive implementations of Bootstrap v4 for Vue.js with extensive and automated WAI-ARIA accessibility markup.

[React Bootstrap](#) is a package of Bootstrap components built with React.

[React Router Bootstrap](#) is a package that provides integration between [React Router](#) v4 and React Bootstrap.

[Bootstrap Table](#) is an extended table to integration with some of the most widely used CSS frameworks.
(Supports Bootstrap, Semantic UI, Bulma, Material Design, Foundation, Vue.js)

[AdminLTE-Laravel](#) is a Laravel 5 package that switchs default Laravel scaffolding/boilerplate to AdminLTE template and Pratt Landing Page with Bootstrap 3.0.

45. Elixir

[Back to the Top](#)



Elixir Guide

A guide covering Elixir language including the applications and tools that will make you a better and more efficient Elixir developer.

Resources

[Elixir](#) is a dynamic, functional language designed for building scalable and maintainable applications. Elixir leverages the [Erlang](#) VM, known for running low-latency, distributed and fault-tolerant systems, while also being successfully used in web development, embedded software, data ingestion, and multimedia processing domains.

[Learning Elixir](#)

[Elixir School](#) is the premier destination for people seeking to learn and master the Elixir programming language.

[Elixir Style Guide](#)

[Getting Started with Elixir from Pluralsight](#)

[Intro to Mix and OTP](#)

[Elixir Programming Language Forum](#)

[The Complete Elixir and Phoenix Bootcamp from Udemy](#)

[Top Elixir Courses from udemy](#)

Tools

[Phoenix Framework](#) is web framework for building rich, interactive web applications quickly, with less code and fewer moving parts. Join our growing community of developers using Phoenix to craft APIs, HTML5 apps and more, for fun or at scale. Note: Phoenix is currently in prototype mode.

[asdf](#) is a CLI tool that can manage multiple language runtime versions with support for Ruby, Node.js, Elixir, Erlang & more.

[CubDB](#) is an embedded key-value database written in the Elixir language. It runs locally, it is schema-less, and backed by a single file.

[Plausible Analytics](#) is a simple, lightweight (< 1 KB), open-source and privacy-friendly alternative to Google Analytics. It doesn't use cookies and is fully compliant with GDPR, CCPA and PECR. You can self-host Plausible or have us run it for you in the Cloud.

[Credo](#) is a static code analysis tool for the Elixir language with a focus on code consistency and teaching.

[Absinthe](#) is a GraphQL toolkit for Elixir.

[Guardian](#) is a token based authentication library for use with Elixir applications.

[Distillery](#) is a tool that simplify deployments in Elixir with OTP releases.

[Quantum](#) is a cron-like job scheduler for Elixir.

[ElixirScript](#) is a tool that converts a subset (or full set) of Elixir code to JavaScript, providing the ability to write JavaScript in Elixir.

[Floki](#) is a simple HTML parser that enables search for nodes using CSS selectors.

[Timex](#) is a complete date/time library for Elixir projects.

[Ueberauth](#) is an Elixir Authentication System for Plug-based Web Applications.

[Manifold](#) is a fast batch message passing between nodes for Erlang/Elixir.

46. Erlang

[Back to the Top](#)



Erlang Guide

A guide covering the Erlang programming language including the applications and tools that will make you a better and more efficient Erlang developer.

Resources

[Erlang](#) is a programming language used to build massively scalable soft real-time systems with requirements on high availability. Some of its uses are in telecoms, banking, e-commerce, computer telephony and instant messaging.

[Erlang Programming Training Modules](#)

[Getting Started With Erlang](#)

[Erlang Reference Manual](#)

[Erlang courses on Udemy](#)

[Erlang courses on Erlang Solutions](#)

[Erlang/Elixir Syntax: A Crash Course](#)

[Erlang Programming Rules and Conventions](#)

Tools

[Erlang LS](#) is a Visual Studio Code Extension for the Erlang Language Server.

[asdf](#) is a CLI tool that can manage multiple language runtime versions on a per-project basis with support for Ruby, Node.js, Elixir, Erlang & more.

[EMQ X Broker](#) is a fully open source, highly scalable, highly available distributed MQTT messaging broker for IoT, M2M and Mobile applications that can handle tens of millions of concurrent clients.

[Lumen](#) is a new compiler and runtime for BEAM languages, designed for WebAssembly.

[Cowboy](#) is a small, fast and modern HTTP server for Erlang/OTP.

[Apache CouchDB](#) is a seamless multi-master syncing database with an intuitive HTTP/JSON API, designed for reliability.

[Ejabberd](#) is a distributed, fault-tolerant technology that allows the creation of large-scale instant messaging applications. The server can reliably support thousands of simultaneous users on a single node and has been designed to provide exceptional standards of fault tolerance.

[Rustler](#) is a library for writing Erlang NIFs in safe Rust code meaning there should be no ways to crash the BEAM (Erlang VM). The library provides facilities for generating the boilerplate for interacting with the BEAM, handles encoding and decoding of Erlang terms, and catches rust panics before they unwind into C.

[Tsung](#) is a high-performance benchmark framework for various protocols including HTTP, XMPP, LDAP, etc.

[Alpaca](#) is a statically typed, strict/eagerly evaluated, functional programming language for the Erlang virtual machine (BEAM).

[Manifold](#) is a fast batch message passing between nodes for Erlang/Elixir.

[Lager](#) is a logging framework for Erlang/OTP.

[Hackney](#) is a simple HTTP client in Erlang.

47. Lua

[Back to the Top](#)



Lua Guide

A guide covering the Lua programming language including the applications and tools that will make you a better and more efficient Lua developer.

Resources

[Lua](#) is a subset of the C programming language designed to be a lightweight embeddable scripting language used by many frameworks and games.

[Programming in Lua ebook](#)

[Getting started with Lua](#)

[Intro to Lua from Core Academy](#)

[Online Lua Classes on Skillshare](#)

[Online Lua Classes on Udemy](#)

[Lua Training Courses from Nobleprog](#)

[RoboCode Academy from Roblox](#)

Tools

[Lua Language Server](#) is an extension for VSCode that provides support for the Lua Language Server.

[Apache APISIX](#) is a dynamic, real-time, high-performance API gateway, based on the Nginx library and etcd.

[NodeMCU](#) is an open source Lua based firmware for the [ESP8266 WiFi SOC from Espressif](#) and uses an on-module flash-based [SPIFFS](#) file system.

[GopherLua](#) is a Lua 5.1 VM and compiler written in Go.

[MoonScript](#) is a programmer friendly language that compiles into Lua.

[Lapis](#) is a web framework for Lua/MoonScript supporting [OpenResty](#) or [http.server](#).

[Algernon](#) is a web server with built-in support for QUIC, HTTP/2, Lua, Markdown, Pongo2, HyperApp, Amber, Sass(SCSS), GCSS, JSX, BoltDB (built-in, stores the database in a file, like SQLite), Redis, PostgreSQL, MariaDB/MySQL, rate limiting, graceful shutdown, plugins, users and permissions.

[Lua Fun](#) is a high-performance functional programming library for Lua designed with LuaJIT's trace compiler in mind.

[Luakit](#) is a fast, light and simple to use micro-browser framework extensible by Lua using the WebKit web content engine and the GTK+ toolkit.

[Lua Resty HTTP](#) is an HTTP client cosocket driver for [OpenResty](#)/ [ngx_lua](#).

[NLua](#) is a bridge between Lua world and the .NET (compatible with .NET Core/UWP/Mac/Linux/Android/iOS/tvOS).

48. Vala

[Back to the Top](#)



Vala Guide

A guide covering the Vala programming language including the applications and tools that will make you a better and more efficient Vala developer.

Resources

[Vala](#) is a programming language that aims to bring modern programming language features to [GNOME](#) developers by using the GObject type system and having additional code generation routines that make targeting the GNOME stack simple. Vala does all this without imposing any additional runtime requirements and without using a different ABI compared to applications and libraries written in C.

[Valadoc](#)

[Vala Reference Manual](#)

[Vala community on GitHub](#)

[Vala community on GitLab](#)

[GTK and Vala](#)

[Getting Started with Vala](#)

[Tutorial for Vala beginners](#)

[Installing Vala on your preferred platform](#)

[Installing Gnome Builder](#)

[GNOME Builder documentation](#)

Tools

[Valac](#) is a Vala compiler that compiles the Vala program in C and uses the C compiler to create a binary file.

[Gnome Builder](#) is an open-source IDE that is written in C and built using the GTK+ toolkit. It has support for the following programming languages: C, C++, Python, Rust and Vala.

[vala-vscode](#) is an extension that provides Vala and Genie integration for Visual Studio such as Syntax highlighting and code intelligence.

[Sequeler](#) is a native Linux SQL client built in Vala and Gtk. It allows you to connect to your local and remote databases, write SQL in a handy text editor with language recognition, and visualize SELECT results in a Gtk.Grid Widget.

[Code](#) is a fast and lightweight code editor designed for elementary OS.

[Granite](#) is a companion library for GTK+ and GLib. Also, it provides complex widgets and convenience functions designed for use in apps built for elementary OS.

[GameHub](#) is a unified library for all your games. It allows users to store all their games from different platforms(Windows, macOS, and Linux) into one program.

49. Haskell

[Back to the Top](#)



Haskell Guide

A guide covering Haskell scripting language including the applications, libraries and tools that will make you a better and more efficient Haskell developer.

Resources

[Haskell](#) is an advanced purely-functional programming language.

[Haskell Documentation](#)

Haskell Community

[Learning Haskell on HaskellWiki](#)

[Haskell Development Libraries and Tools](#)

[Learning Haskell Programming on Udemy](#)

[Introduction to Functional Programming with Haskell on edX](#)

[Haskell Fundamentals course on Pluralsight](#)

[Functional Programming in Haskell](#)

Tools

[Haskell Dockerfile Linter](#) is a smarter Dockerfile linter that helps you build [best practice](#) Docker images.

[ShellCheck](#) is a shell script static analysis tool.

[Haskell for Visual Studio Code](#) is an extension adds language support for Haskell, powered by the [Haskell Language Server](#).

[Haskelly](#) is a VS Code extension that provides complete support for casual and expert Haskell development.

[Miso](#) is a small, production-ready, Haskell front-end framework for quickly building highly interactive single-page web applications. It features a virtual-dom, recursive diffing / patching algorithm, attribute and property normalization, event delegation, event batching, SVG, Server-sent events, Websockets, type-safe [servant](#)-style routing and an extensible Subscription-based subsystem.

[Grenade](#) is a composable, dependently typed, and fast recurrent neural network library for Machine Learning in Haskell.

[IntelliJ Haskell](#) is an IntelliJ plugin for Haskell.

[CodeWorld](#) is an educational environment using Haskell. It provides a simple mathematical model for geometric figures, animations, and interactive and multi-player games.

50. PHP

[Back to the Top](#)



PHP Guide

A guide covering PHP including the applications and tools that will make you a better and more efficient PHP developer.

Resources

[PHP](#) is a popular general-purpose scripting language that is especially suited to web development. Fast, flexible and pragmatic, PHP powers everything from your blog to the most popular websites in the world.

[PHP Manual](#)

[MIT's PHP Code Style Guide](#)

[PHP Style Guide](#)

[PHP 7 Tutorial - An Ultimate Guide for Beginners](#)

[PHP tutorial by W3Schools](#)

[PHP MySQL & CodeIgniter Course by Udemy](#)

Tools

[PhpStorm](#) is a professional PHP IDE developed by Jet Brains for working with Symfony, Laravel, Drupal, WordPress, Laminas, Magento, Joomla!, CakePHP, Yii, and other frameworks.

[Laravel](#) is a web application framework with expressive, elegant syntax. We believe development must be an enjoyable and creative experience to be truly fulfilling.

[PHP Tools for Visual Studio/VS Code](#) is a an extension that turn Visual Studio into a powerful PHP development environment.

[Symfony](#) is a PHP framework for web and console applications and a set of reusable PHP components. Symfony is used by thousands of web applications (including BlaBlaCar.com and Spotify.com) and most of the [popular PHP projects](#) (including Drupal and Magento).

[CakePHP](#) is a rapid development framework for PHP which uses commonly known design patterns like Associative Data Mapping, Front Controller, and MVC. CakePHP's main goal is to provide a structured framework that enables PHP users at all levels to rapidly develop robust web applications, without any loss to flexibility.

[Composer](#) is a tools helps you declare, manage, and install dependencies of PHP projects.

[Guzzle](#) is a PHP HTTP client that makes it easy to send HTTP requests and trivial to integrate with web services.

[DesignPatternsPHP](#) is a collection of known design patterns and some sample code how to implement them in PHP 7.4. Every pattern has a small list of examples.

[CodeIgniter](#) is an Application Development Framework for people who build web sites using PHP. Its goal is to enable you to develop projects much faster than you could if you were writing code from scratch, by providing a rich set of libraries for commonly needed tasks, as well as a simple interface and logical structure to access these libraries. CodeIgniter lets you creatively focus on your project by minimizing the amount of code needed for a given task.

[HHVM](#) is an open-source virtual machine designed for executing programs written in [Hack](#). HHVM uses a just-in-time (JIT) compilation approach to achieve superior performance while maintaining amazing development flexibility. HHVM should be used together with a webserver like the built in, easy to deploy [Proxygen](#), or a [FastCGI-based](#) webserver on top of nginx or Apache.

[PHPUnit](#) is a programmer-oriented testing framework for PHP. It is an instance of the xUnit architecture for unit testing frameworks.

[Phalcon](#) is an open source web framework delivered as a C extension for the PHP language providing high performance and lower resource consumption.

[Swoole](#) is an event-driven asynchronous & coroutine-based concurrency networking communication engine with high performance written in C and C++ for PHP.

[Matomo](#) is a full-featured PHP MySQL software program that you download and install on your own webserver. At the end of the five-minute installation process, you will be given a JavaScript code. Simply copy and paste this tag on websites you wish to track and access your analytics reports in real-time.

[Grav](#) is a Fast, Simple, and Flexible, file-based Web-platform. There is Zero installation required. Just extract the ZIP archive, and you are already up and running. It follows similar principles to other flat-file CMS platforms, but has a different design philosophy than most. Grav comes with a powerful Package Management System to allow for simple installation and upgrading of plugins and themes, as well as simple updating of Grav itself.

[Whoops](#) is an error handler framework for PHP. Out-of-the-box, it provides a pretty error interface that helps you debug your web projects, but at heart it's a simple yet powerful stacked error handling system.

[Slim](#) is a PHP micro framework that helps you quickly write simple yet powerful web applications and APIs.

51. Robotics

[Back to the Top](#)



Robotics Guide

A guide covering Robotics including the applications, libraries and tools that will make you a better and more efficient Robotics developer.

AWS DeepRacer autonomous Car Kit



[Checkout the AWS DeepRacer autonomous Car Kit](#)

AWS DeepRacer autonomous Car Kit Hardware Specifications

- CAR: 18th scale 4WD with monster truck chassis
- CPU: Intel Atom™ Processor
- MEMORY: 4GB RAM
- STORAGE: 32GB (expandable)
- WI-FI: 802.11ac
- CAMERA: 4 MP cameras with MJPEG
- SOFTWARE: Ubuntu OS 16.04 LTS, Intel® OpenVINO™ toolkit, ROS Kinetic
- DRIVE BATTERY: 7.4V/1100mAh lithium polymer
- COMPUTE BATTERY: 13600mAh USB-C PD
- PORTS: 4x USB-A, 1x USB-C, 1x Micro-USB, 1x HDMI
- SENSORS: Integrated accelerometer and gyroscope

AWS DeepRacer Evo autonomous Car Kit

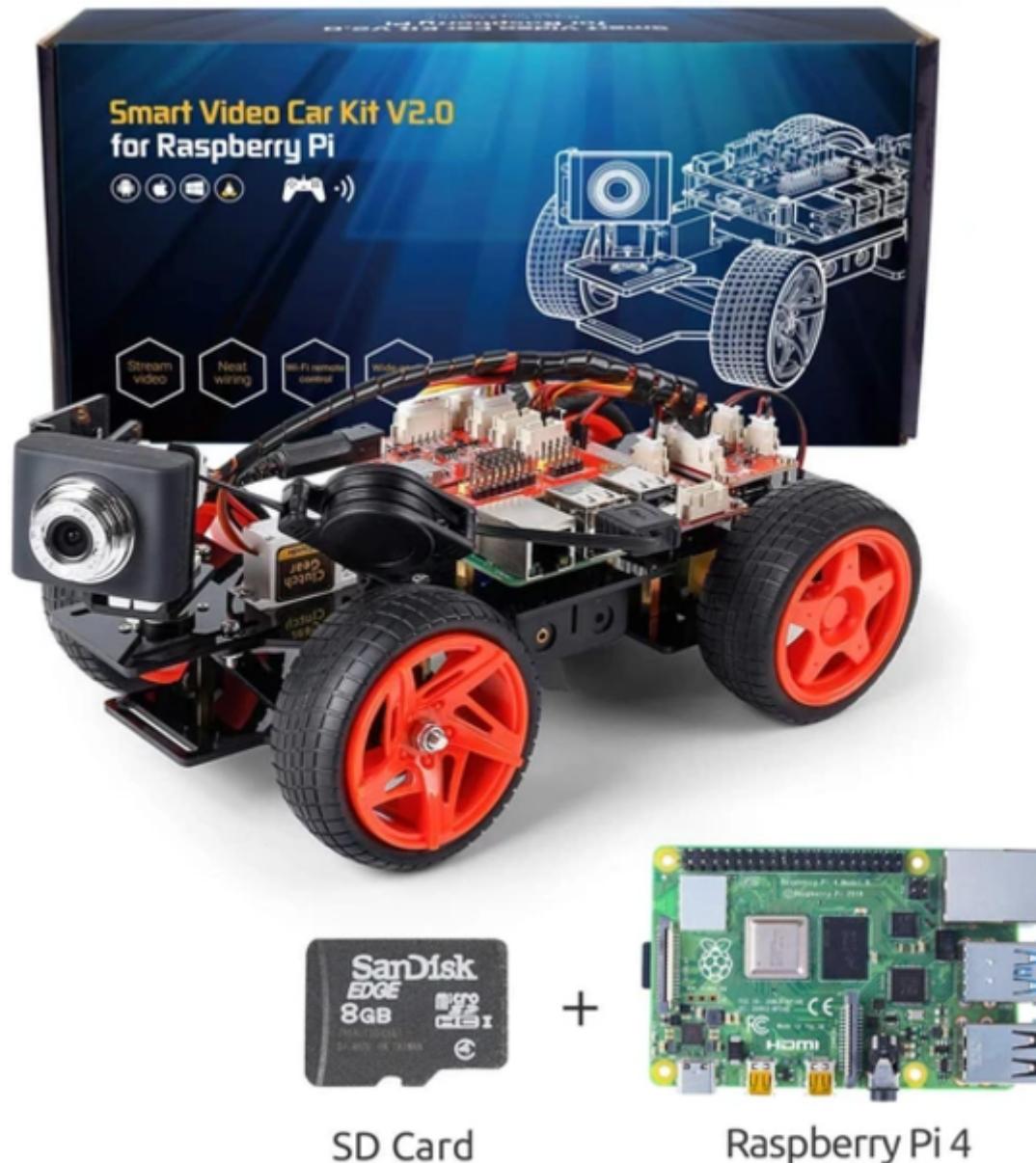


[Checkout the AWS DeepRacer Evo autonomous Car Kit](#)

AWS DeepRacer Evo autonomous Car Kit Hardware Specifications

- CAR: 18th scale 4WD with monster truck chassis
- CPU: Intel Atom™ Processor
- MEMORY: 4GB RAM
- STORAGE: 32GB (expandable)
- WI-FI: 802.11ac
- CAMERA: Stereo 4 MP cameras with MJPEG
- **LIDAR Sensor: 360 Degree 12 Meters Scanning Radius LIDAR Sensor**
- SOFTWARE: Ubuntu OS 16.04 LTS, Intel® OpenVINO™ toolkit, ROS Kinetic
- DRIVE BATTERY: 7.4V/1100mAh lithium polymer
- COMPUTE BATTERY: 13600mAh USB-C PD
- PORTS: 4x USB-A, 1x USB-C, 1x Micro-USB, 1x HDMI
- SENSORS: Integrated accelerometer and gyroscope

[Checkout the SunFounder PiCar-V Kit V2.0 for Raspberry Pi](#)



Resources

[Robotics courses from Coursera](#)

[Learn Robotics with Online Courses and Classes from edX](#)

[Top Robotics Courses Online from Udemy](#)

[Free Online AI & Robotics Courses](#)

[REC Foundation Robotics Industry Certification](#)

[Carnegie Mellon Robotics Academy](#)

[RIA Robotic Integrator Certification Program](#)

[AWS RoboMaker – Develop, Test, Deploy, and Manage Intelligent Robotics Apps](#)

[Microsoft AI School](#)

[Language Understanding \(LUIS\) for Azure Cognitive Services](#)

[ROS on Windows 10](#)

[Windows ML ROS Node](#)

[Azure VM templates to bootstrap ROS and ROS 2 environments](#)

[Google Robotics Research](#)

Tools

[Robot Framework](#) is a generic open source automation framework. It can be used for test automation and robotic process automation. It has easy syntax, utilizing human-readable keywords. Its capabilities can be extended by libraries implemented with Python or Java.

[The Robotics Library \(RL\)](#) is a self-contained C++ library for robot kinematics, motion planning and control. It covers mathematics, kinematics and dynamics, hardware abstraction, motion planning, collision detection, and visualization. RL runs on many different systems, including Linux, macOS, and Windows. It uses CMake as a build system and can be compiled with Clang, GCC, and Visual Studio.

[ROS](#) is robotics middleware. Although ROS is not an operating system, it provides services designed for a heterogeneous computer cluster such as hardware abstraction, low-level device control, implementation of commonly used functionality, message-passing between processes, and package management.

[ROS2](#) is a set of [software libraries and tools](#) that help you build robot applications. From drivers to state-of-the-art algorithms, and with powerful developer tools, ROS has what you need for your next robotics project. And it's all open source.

[MoveIt](#) is the most widely used software for manipulation and has been used on over 100 robots. It provides an easy-to-use robotics platform for developing advanced applications, evaluating new designs and building integrated products for industrial, commercial, R&D, and other domains.

[AutoGluon](#) is toolkit for [Deep learning](#) that automates machine learning tasks enabling you to easily achieve strong predictive performance in your applications. With just a few lines of code, you can train and deploy high-accuracy deep learning models on tabular, image, and text data.

[Gazebo](#) accurately and efficiently simulates indoor and outdoor robots. You get a robust physics engine, high-quality graphics, and programmatic and graphical interfaces.

[Robotics System Toolbox](#) provides tools and algorithms for designing, simulating, and testing manipulators, mobile robots, and humanoid robots. For manipulators and humanoid robots, the toolbox includes algorithms for collision checking, trajectory generation, forward and inverse kinematics, and dynamics using a rigid body tree representation. For mobile robots, it includes algorithms for mapping, localization, path planning, path following, and motion control. The toolbox provides reference examples of common industrial robot applications. It also includes a library of commercially available industrial robot models that you can import, visualize, and simulate.

[Intel Robot DevKit](#) is the tool to generate Robotics Software Development Kit (RDK) designed for autonomous devices, including the ROS2 core and capabilities packages like perception, planning, control driver etc. It provides flexible build/runtime configurations to meet different autonomous requirement on top of diversity hardware choices, for example use different hardware engine CPU/GPU/VPU to accelerate AI related features.

[Arduino](#) is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board.

[ArduPilot](#) enables the creation and use of trusted, autonomous, unmanned vehicle systems for the peaceful benefit of all. ArduPilot provides a comprehensive suite of tools suitable for almost any vehicle and application.

[AirSim](#) is a simulator for drones, cars and more, built on Unreal Engine (we now also have an experimental Unity release). It is open-source, cross platform, and supports hardware-in-loop with popular flight controllers such as PX4 for physically and visually realistic simulations.

The [JPL Open Source Rover](#) is an open source, build it yourself, scaled down version of the 6 wheel rover design that JPL uses to explore the surface of Mars. The Open Source Rover is designed almost entirely out of consumer off the shelf (COTS) parts. This project is intended to be a teaching and learning experience for those who want to get involved in mechanical engineering, software, electronics, or robotics.

[Light Detection and Ranging\(LiDAR\)](#) is a remote sensing method that uses light in the form of a pulsed laser at an object, and uses the time and wavelength of the reflected beam of light to estimate the distance and in some applications ([Laser Imaging](#)), to create a 3D representation of the object and its surface characteristics. This technology is commonly used in aircraft and self-driving vehicles.

[AliceVision](#) is a Photogrammetric Computer Vision Framework which provides a 3D Reconstruction and Camera Tracking algorithms. AliceVision aims to provide strong software basis with state-of-the-art computer vision algorithms that can be tested, analyzed and reused. The project is a result of collaboration between academia and industry to provide cutting-edge algorithms with the robustness and the quality required for production usage.

[CARLA](#) is an open-source simulator for autonomous driving research. CARLA has been developed from the ground up to support development, training, and validation of autonomous driving systems. In addition to open-source code and protocols, CARLA provides open digital assets (urban layouts, buildings, vehicles) that were created for this purpose and can be used freely. The simulation platform supports flexible specification of sensor suites and environmental conditions.

[ROS bridge](#) is a package to bridge ROS for CARLA Simulator.

[ROS-Industrial](#) is an open source project that extends the advanced capabilities of ROS software to manufacturing.

[AWS RoboMaker](#) is the most complete cloud solution for robotic developers to simulate, test and securely deploy robotic applications at scale. RoboMaker provides a fully-managed, scalable infrastructure for simulation that customers use for multi-robot simulation and CI/CD integration with regression testing in simulation.

[Microsoft Robotics Developer Studio](#) is a free .NET-based programming environment for building robotics applications.

[Visual Studio Code Extension for ROS](#) is an extension provides support for Robot Operating System (ROS) development.

[Azure Kinect ROS Driver](#) is a node which publishes sensor data from the [Azure Kinect Developer Kit](#) to the [Robot Operating System \(ROS\)](#). Developers working with ROS can use this node to connect an Azure Kinect Developer Kit to an existing ROS installation.

[Azure IoT Hub for ROS](#) is a ROS package works with the Microsoft Azure IoT Hub service to relay telemetry messages from the Robot to Azure IoT Hub or reflect properties from the Digital Twin to the robot using dynamic reconfigure.

[ROS 2 with ONNX Runtime](#) is a program that uses ROS 2 to run on different hardware platforms using their respective AI acceleration libraries for optimized execution of the ONNX model.

[Azure Cognitive Services LUIS ROS Node](#) is a ROS node that bridges between ROS and the Azure Language Understanding Service. it can be configured to process audio directly from a microphone, or can subscribe to a ROS audio topic, then processes speech and generates "intent" ROS messages which can be processed by another ROS node to generate ROS commands.

52. WSL

[Back to the Top](#)



WSL Guide

A guide for using WSL and all the tools/utilities that will make you a better and more efficient WSL developer.

Resources

[WSL 2 Linux Kernel on GitHub](#) is the source for the Linux kernel used in Windows Subsystem for Linux 2 (WSL2).

[WSLConf](#) is a community-initiated event on all things Windows Subsystem for Linux and WSL-related.

[Comparing WSL 2 and WSL 1](#)

[GPU accelerated machine learning training in the Windows Subsystem for Linux](#)

[CUDA on Windows Subsystem for Linux\(WSL\) 2](#)

[Developing in Windows Subsystem for Linux \(WSL\)](#)

[Using WSL 2 with Visual Studio Code](#)

[WSLG: X11 and Wayland Applications in WSL](#)

[How to run Podman on Windows with WSL 2](#)

[Getting started with MySQL, MongoDB, PostgreSQL, SQLite, Microsoft SQL Server, or Redis to set up a database on WSL](#)

[Setting up SAP HANA, express edition into WSL 2 \(Windows Subsystem for Linux\)](#)

[Set up your Node.js development environment with WSL 2](#)

[Getting started mounting a Linux disk in WSL 2](#)

[Using Fedora with Microsoft's WSL 2](#)

Tools

[wslu](#) is a collection of utilities for Windows 10 Linux Subsystem, such as retrieving Windows 10 environment variables or creating your favorite Linux GUI application shortcuts on Windows 10 Desktop.

[Ubuntu on WSL](#) is a wiki guide on getting started with the latest version of Ubuntu installed and setup on WSL for Windows 10.

[Ubuntu Pro for Azure](#) is a premium image designed by Canonical optimized for production environments running on Azure. It includes security and compliance services, enabled by default, in a form suitable for small to large-scale Linux enterprise operations — with no contract needed.

[Azure CLI](#) is a set of commands used to create and manage Azure resources. The Azure CLI is available across Azure services and is designed to get you working quickly with Azure, with an emphasis on automation.

[Visual Studio Code Remote - WSL extension](#) lets you use the Windows Subsystem for Linux (WSL) as your full-time development environment right from VS Code. You can develop in a Linux-based environment, use Linux-specific toolchains and utilities, and run and debug your Linux-based applications all from the comfort of Windows. The extension runs commands and other extensions directly in WSL so you can edit files located in WSL or the mounted Windows filesystem (for example /mnt/c) without worrying about pathing issues, binary compatibility, or other cross-OS challenges.

[Windows Terminal](#) is a new, modern, feature-rich, productive terminal application for command-line users. It includes many of the features most frequently requested by the Windows command-line community including support for tabs, rich text, globalization, configurability, theming & styling, and more.

[PowerShell Core](#) is a cross-platform (Windows, Linux, and macOS) automation and configuration tool/framework that works well with your existing tools and is optimized for dealing with structured data (e.g. JSON, CSV, XML, etc.), REST APIs, and object models. It includes a command-line shell, an associated scripting language and a framework for processing cmdlets.

[Docker Desktop WSL 2 backend](#) creates an architectural change that gives a full Linux kernel built by Microsoft, allowing Linux containers to run natively without emulation. With Docker Desktop running on WSL 2, users can leverage Linux workspaces and avoid having to maintain both Linux and Windows build scripts. In addition, WSL 2 provides improvements to file system sharing, boot time, and allows access to some cool new features for Docker Desktop users.

[Dxgkrnl](#) is a brand-new kernel driver for Linux that exposes the /dev/dxg device to user mode Linux. /dev/dxg exposes a set of IOCTL that closely mimic the native WDDM D3DKMT kernel service layer on

Windows. Dxgkrnl inside of the Linux kernel connects over the VM Bus to its big brother on the Windows host and uses this VM bus connection to communicate with the physical GPU.

[Ansible-WSL](#) is an open source program that makes it easier to provision your Windows from inside of WSL by Ansible.

[WSL-DistroLauncher](#) is a sample/reference launcher app for WSL distro Microsoft Store packages.

[Pengwin](#) is a Linux distro optimized for WSL based on Debian.

[Pengwin Enterprise](#) is an enterprise Linux solution for Windows Subsystem for Linux (WSL) that is compatible with mainstream enterprise Linux distributions.

53. 3D Graphics & Design

[Back to the Top](#)



3D Graphics and Design Guide

A guide covering 3D Graphics and Design including the applications, libraries and tools that will make you a better and more efficient Graphics Designer.

Resources

[Introduction to Pixar's Universal Scene Description \(USD\)](#)

[Intro to using Universal Scene Description with NVIDIA](#)

[First steps with Universal Scene Description in Blender](#)

[AMD Radeon ProRender Developer Suite](#)

[Blender Foundation](#)

[Blender Foundation Certification Training](#)

[Blender Cloud Courses](#)

[Blender Institute](#)

[Blender Education](#)

[Blender Network](#)

[Blender Courses From Udemy](#)

[AutoDesk Learning, Training & Certification](#)

[AutoDesk Design Academy](#)

[AutoDesk Courses and Specializations from Coursera](#)

[Cinema 4D Quick Tips](#)

[Getting Started with Cinema 4D](#)

[Graphic Design Masterclass\(Photoshop, Illustrator, InDesign\) From Udemy](#)

[Vectr: Beginner's Guide To Graphic Design from udemy](#)

[Canva MasterClass: Design For EveryDay Use from Udemy](#)

Tools

[Adobe Creative Cloud](#) is a collection of 20+ desktop and mobile apps(Lightroom, Photoshop, Illustrator, InDesign, Rush, Etc.) and services for photography, design, video, web, UX, and more.

[AutoDesk](#) is a collection of professional software products and services used in 3D design, architecture, engineering, manufacturing, and high production entertainment companies.

[AutoDesk Maya](#) is a professional 3D computer animation software with powerful modeling, rendering, simulation, texturing, and animation tools for artists, modelers, and animators.

[AutoDesk 3ds Max](#) is a professional software program for 3D modeling, animation, rendering, and visualization. 3ds Max allows you to create stunning game environments, design visualizations, and virtual reality experiences.

[Blender](#) is a professional, free and open-source 3D computer graphics software toolset used for creating animated films, visual effects, art, 3D printed models, interactive 3D applications and video games.

[Cinema 4D](#) is a professional 3D and VFX software suite that is capable of doing procedural and polygonal modeling, animation, lighting, texturing, motion graphic and rendering.

[Houdini](#) is a 3D procedural software for modeling, rigging, animation, VFX, look development, lighting and rendering in film, TV, advertising and video game pipelines.

[Open Graphics Library\(OpenGL\)](#) is an API used acrossed mulitple programming languages and platforms for hardware-accelerated rendering of 2D/3D vector graphics currently developed by the [Khronos Group](#).

[Open Computing Language \(OpenCL\)](#) is an open standard for [parallel programming](#) of heterogeneous platforms consisting of CPUs, GPUs, and other hardware accelerators found in supercomputers, cloud servers, personal computers, mobile devices and embedded platforms.

[Vulkan](#) is a modern cross-platform graphics and compute API that provides high-efficiency, cross-platform access to modern GPUs used in a wide variety of devices from PCs and consoles to mobile phones and embedded platforms. Vulkan is currently in development by the Khronos consortium.

[Metal](#) is a low-level GPU programming framework used for rendering 2D and 3D graphics on Apple platforms such as iOS, iPadOS, macOS, watchOS and tvOS.

[MoltenVK](#) is an implementation of Vulkan running on iOS and macOS using Apple's [Metal](#) graphics framework.

[MoltenGL](#) is an implementation of the OpenGL ES 2.0 API that runs on Apple's [Metal](#) graphics framework.

[AMD Radeon ProRender](#) is a powerful physically-based rendering engine that enables creative professionals to produce stunningly photorealistic images on virtually any GPU, any CPU, and any OS in over a dozen leading digital content creation and CAD applications.

[NVIDIA Omniverse](#) is a powerful, multi-GPU, real-time simulation and collaboration platform for 3D production pipelines based on Pixar's Universal Scene Description and NVIDIA RTX.

[Universal Scene Description](#) is an efficient, scalable system for authoring, reading, and streaming time-sampled scene description for interchange between graphics applications.

[OpenTimelineIO](#) is an interchange format and API for editorial cut information. OTIO is not a container format for media, rather it contains information about the order and length of cuts and references to external media. OTIO includes both a file format and an API for manipulating that format. It also includes a plugin architecture for writing adapters to convert from/to existing editorial timeline formats. It also implements a dependency- less library for dealing strictly with time, opentime.

[OpenSubdiv](#) is a set of open source libraries that implement high performance subdivision surface (subdiv) evaluation on massively parallel CPU and GPU architectures. This codepath is optimized for drawing deforming subdivs with static topology at interactive framerates. The resulting limit surface matches Pixar's Renderman to numerical precision.

[Affinity Designer](#) is a professional vector graphics editor used by many industry leading illustrators, designers, and other creatives looking to create high-quality concept art.

[Sketch](#) is a professional vector graphics app with a beautiful UI and powerful toolset.

[Krita](#) is an open source professional painting program for digital painting and 2D animation.

[Vectr](#) is a free graphics software used to create vector graphics easily and intuitively.

[Glimpse](#) is a cross-platform raster graphics editor based on the GNU Image Manipulation Program available for Linux, macOS, and Windows. A great tool for making YouTube video thumbnails.

[GNU Image Manipulation Program\(GIMP\)](#) is a cross-platform raster graphics editor based on the GNU Image Manipulation Program(GIMP) available for Linux, macOS, and Windows. A great free alternative for Photoshop.

[PhotoGIMP](#) is a simple Patch for [GIMP 2.10+](#) to help all Adobe Photoshop Users feel at home.

[Photopea](#) is an advanced online photo editor supporting Adobe Photoshop(PSD), GIMP(XCF), Sketch App(Sketch), Adobe XD(XD) and CorelDRAW(CDR) formats.

[Inkscape](#) is an open-source vector graphics editor; it can be used to create or edit vector graphics such as illustrations, diagrams, line arts, charts, logos and complex paintings.

[Gthree](#) is a GObject/Gtk+ port of three.js.

[FreeCAD](#) is a free and open-source general-purpose parametric 3D CAD modeler and a building information modeling software with finite-element-method support.

[Ultimaker Cura](#) is the world's most popular 3D printing software. Prepare prints with a few clicks, integrate with CAD software for an easier workflow, or dive into custom settings for in-depth control.

[Dust3D](#) is a cross-platform open-source modeling software. It helps you create a 3D watertight model in seconds. Use it to speed up your character modeling in game making, 3D printing, and so on.

[Goxel](#) is a free and Open Source 3D Voxel Editor.

[Sketchfab](#) gives you open-source 3D models ready to view, buy, and download for free.

54. Game Development

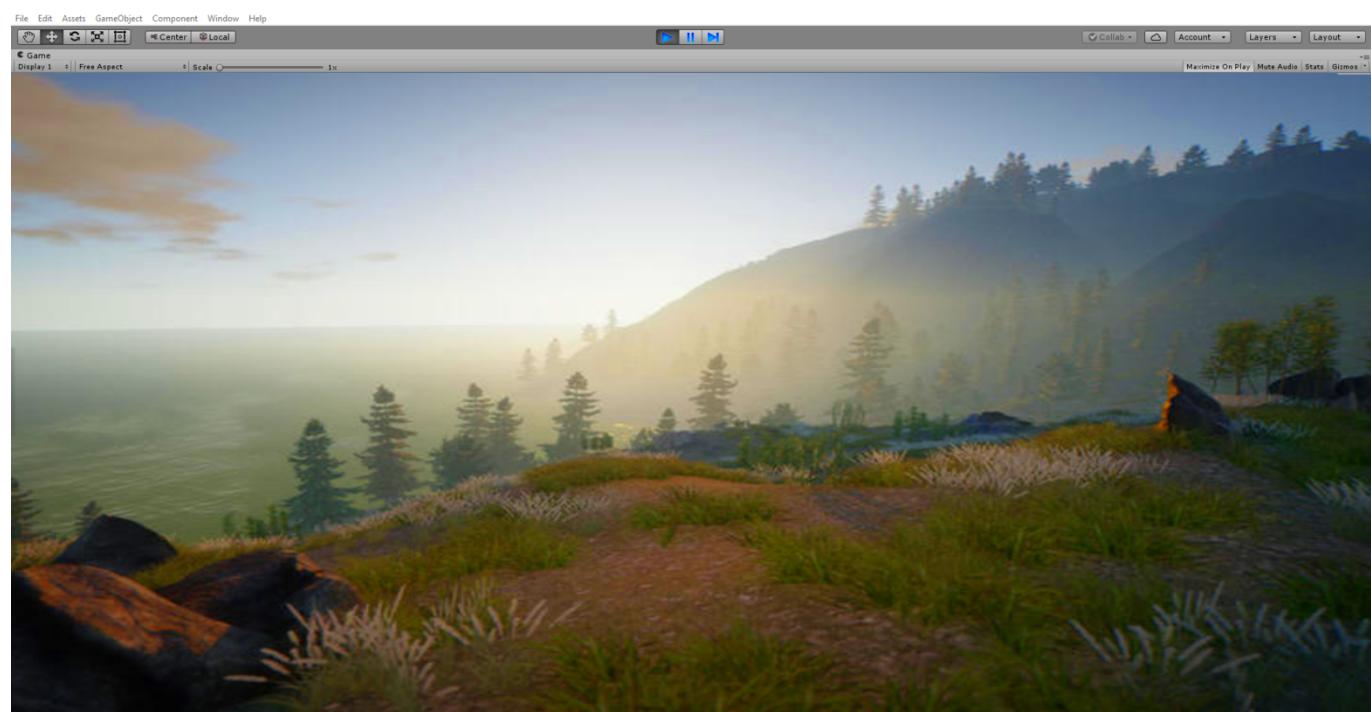
[Back to the Top](#)



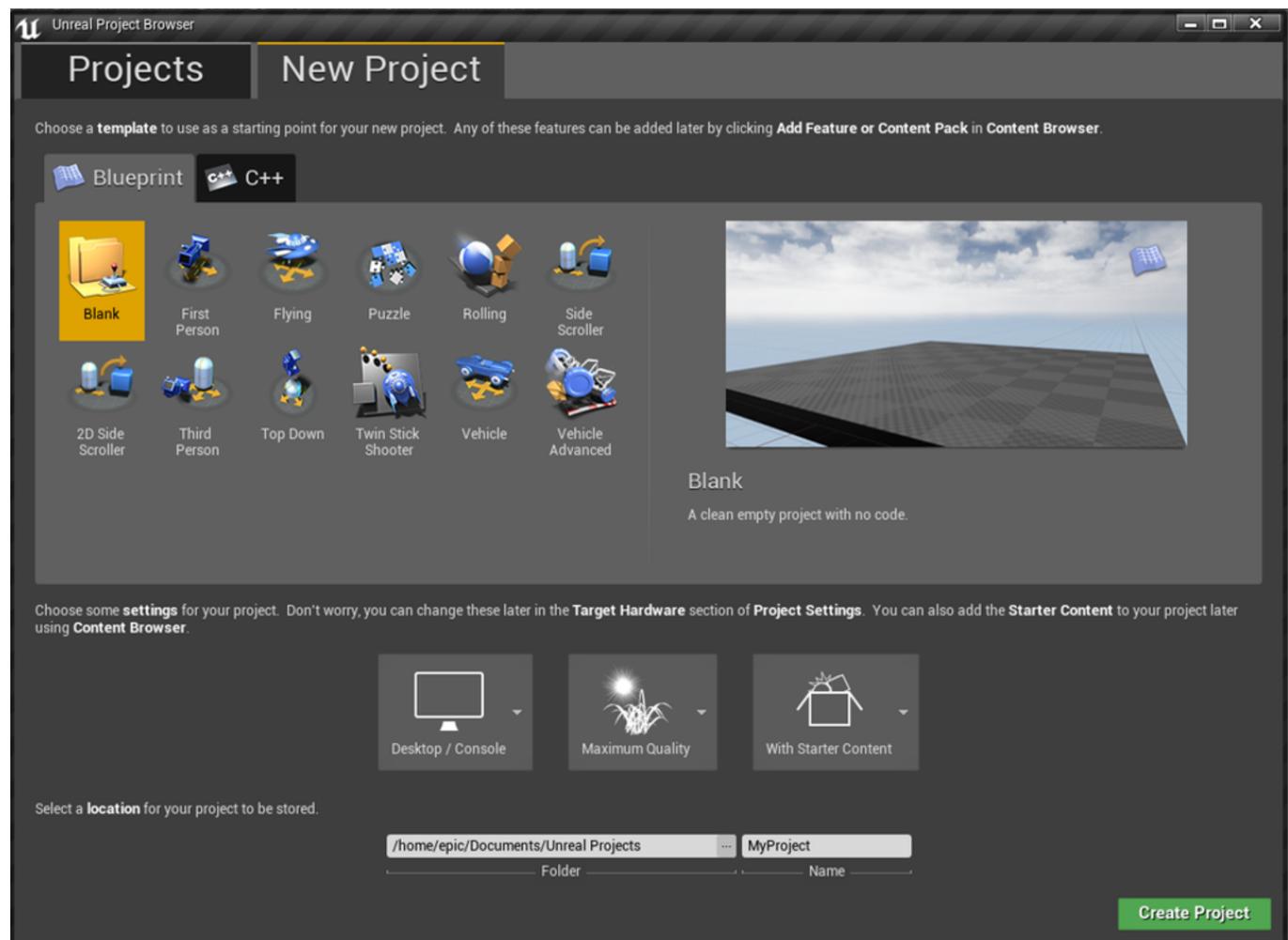
Game Development Guide

A guide for getting started in game development showcasing the best Game Engines, Software, and Libraries for building Games and Simulators.

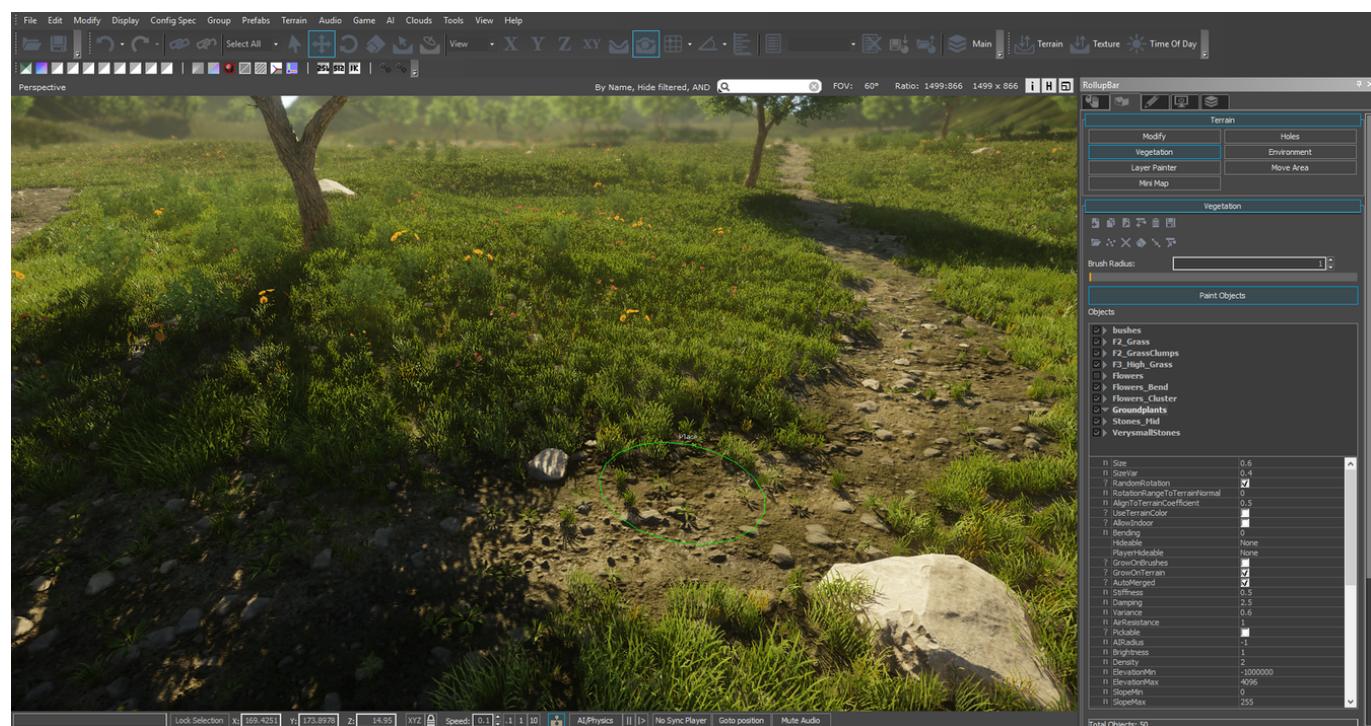
[Checkout the Unity Engine](#)



Checkout the Unreal Engine 4

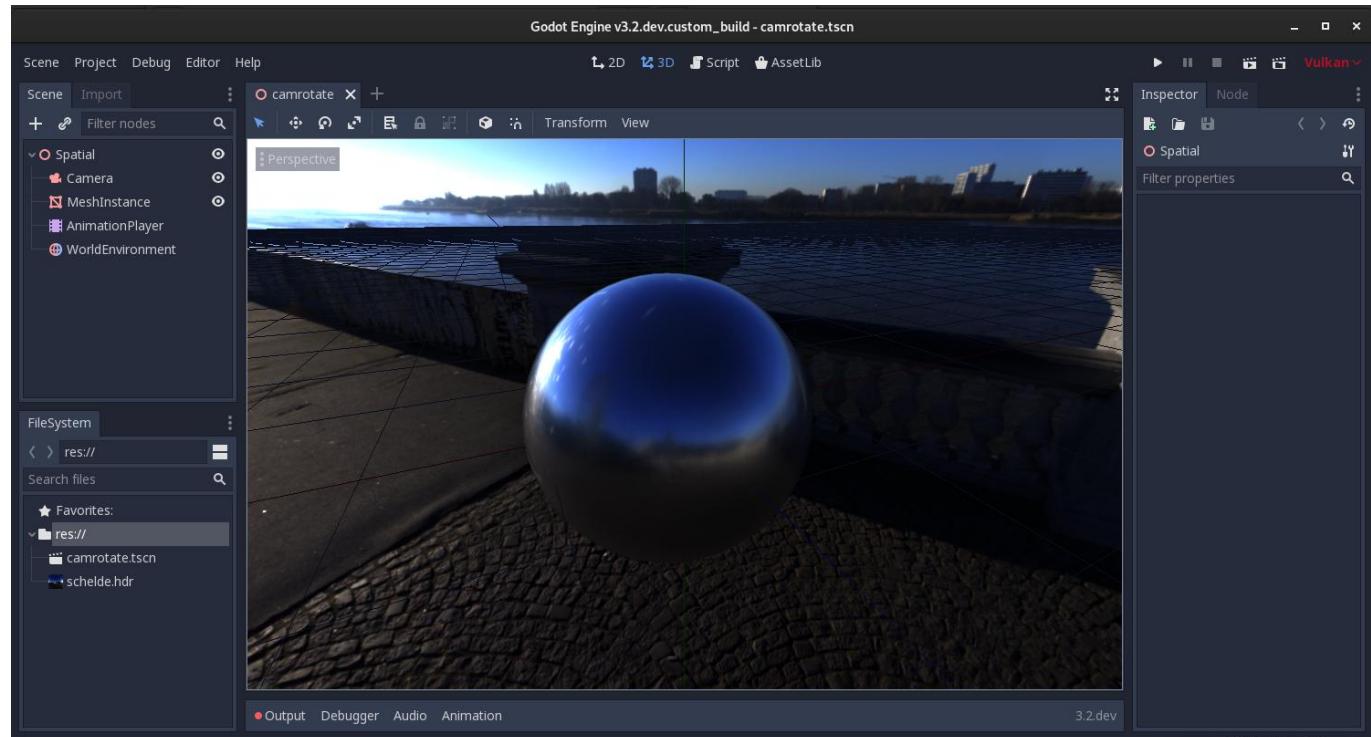


Checkout the CryEngine



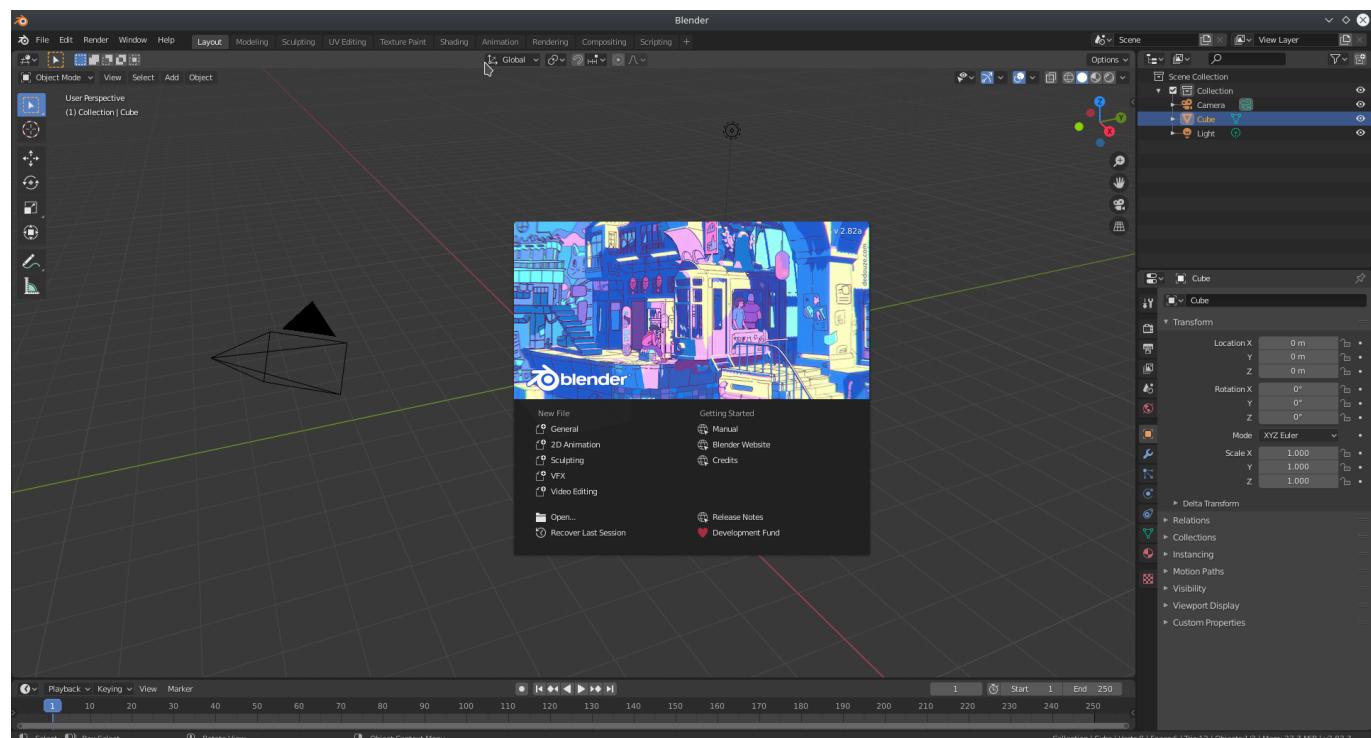
Checkout the Godot Engine

If you would like to Donate to the Godot Project

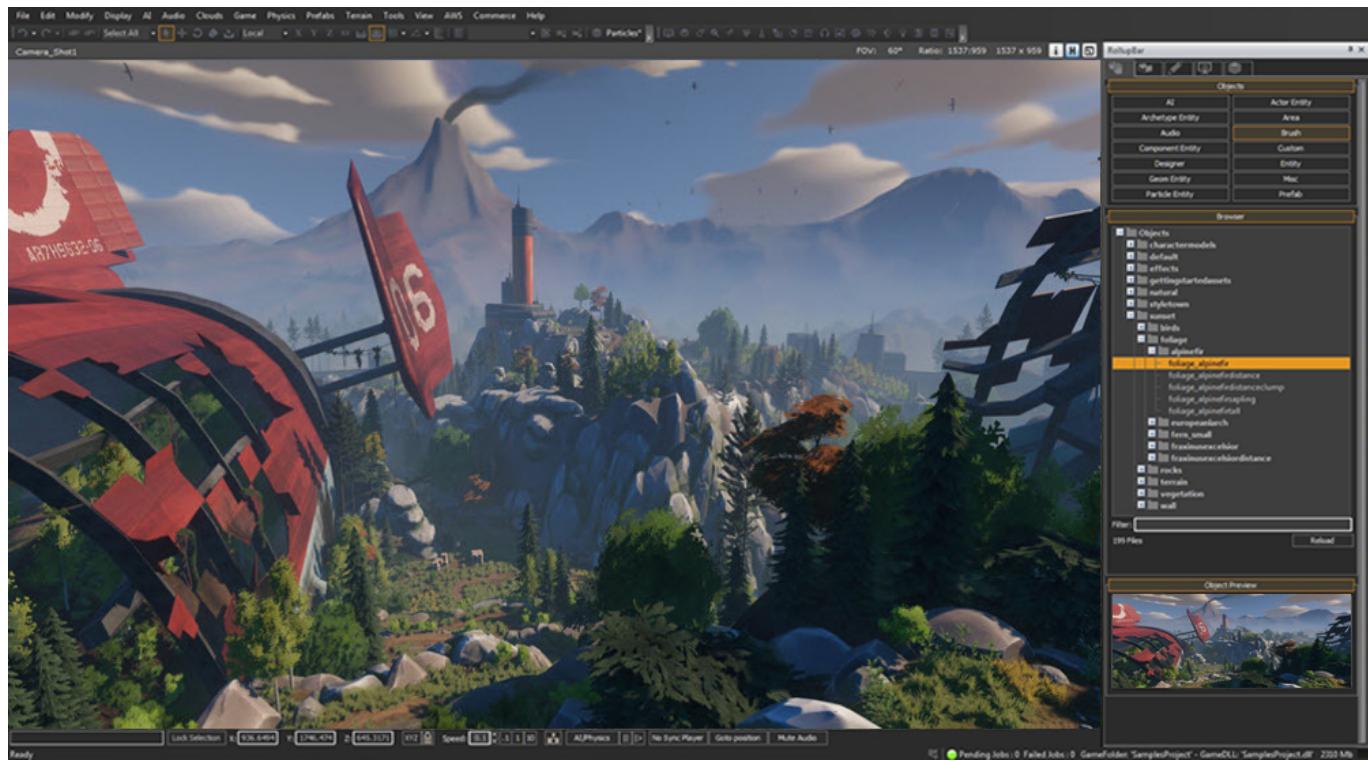


Checkout Blender

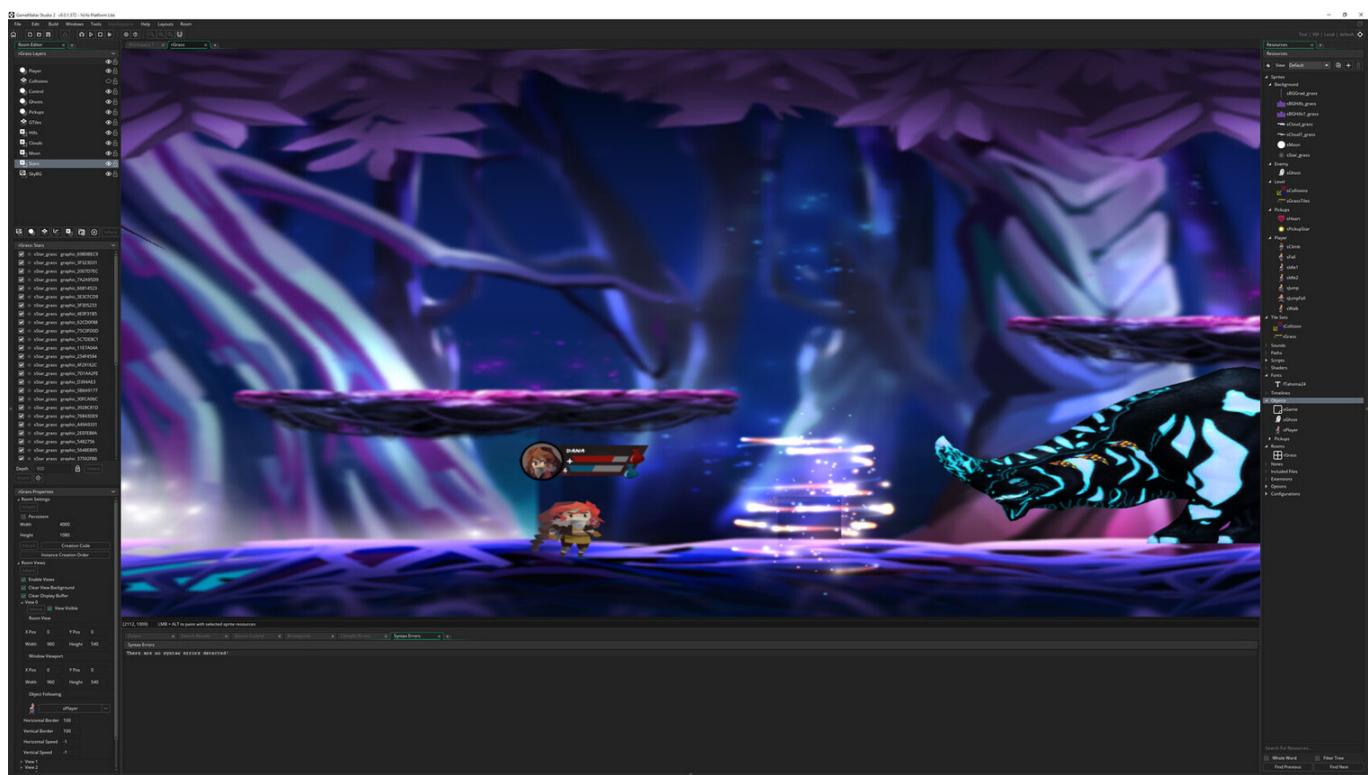
If you would like to Donate to the Blender Project



Checkout AWS Lumberyard(based on CryEngine)



Checkout Game Maker Studio 2



Resources

[Unreal Online Learning](#) is a free learning platform that offers hands-on video courses and guided learning paths.

[Unreal Engine Authorized Training Program](#)

[Unreal Engine for education](#)

[Unreal Engine Training & Simulation](#)

[Unity Certifications](#)

[Autodesk for Games](#)

[Getting Started with DirectX 12 Ultimate](#)

[Getting Started with Vulkan](#)

[Getting Started with Apple Metal](#)

[Game Design Online Courses from Udemy](#)

[Game Design Online Courses from Skillshare](#)

[Learn Game Design with Online Courses and Classes from edX](#)

[Game Design Courses from Coursera](#)

[Game Design and Development Specialization Course from Coursera](#)

Tools

[Unreal Engine](#) is a game engine developed by Epic Games with the world's most open and advanced real-time 3D creation tool. Continuously evolving to serve not only its original purpose as a state-of-the-art game engine, today it gives creators across industries the freedom and control to deliver cutting-edge content, interactive experiences, and immersive virtual worlds.

[Unity](#) is a cross-platform game development platform. Use Unity to build high-quality 3D and 2D games, deploy them across mobile, desktop, VR/AR, consoles or the Web, and connect with loyal and enthusiastic players and customers.

[Unigine](#) is a cross-platform game engine designed for development teams (C++/C# programmers, 3D artists) working on interactive 3D apps.

[Panda3D](#) is a game engine, a framework for 3D rendering and game development for Python and C++ programs, developed by Disney and CMU. Panda3D is open-source and free for any purpose, including commercial ventures.

[Source 2](#) is a 3D video game engine in development by Valve as a successor to Source. It is used in Dota 2, Artifact, Dota Underlords, parts of The Lab, SteamVR Home, and Half-Life: Alyx.

[AutoDesk 3ds Max](#) is a professional software program for 3D modeling, animation, rendering, and visualization. 3ds Max allows you to create stunning game environments, design visualizations, and virtual reality experiences.

[Houdini](#) is a 3D procedural software for modeling, rigging, animation, VFX, look development, lighting and rendering in film, TV, advertising and video game pipelines.

[A-Frame](#) is a web framework for building virtual reality experiences in WebVR with HTML and Entity-Component. A-Frame works on Vive, Rift, desktop, mobile platforms.

[ARKit](#) is a set of software development tools to enable developers to build augmented-reality apps for iOS developed by Apple. The latest version ARKit 3.5 takes advantage of the new LiDAR Scanner and depth sensing system on iPad Pro(2020) to support a new generation of AR apps that use Scene Geometry for enhanced scene understanding and object occlusion.

[RealityKit](#) is a framework to implement high-performance 3D simulation and rendering with information provided by the ARKit framework to seamlessly integrate virtual objects into the real world.

[SceneKit](#) is a high-level 3D graphics framework that helps you create 3D animated scenes and effects in your iOS apps.

[ARCore](#) is a software development kit developed by Google that allows for augmented reality applications in the real world. These tools include environmental understanding, which allows devices to detect horizontal and vertical surfaces and planes. It also includes motion tracking, which lets phones understand and track their positions relative to the world. Also ARCore's Light Estimation API lets your digital objects appear realistically as if they're actually part of the physical world.

[AppGameKit](#) is a powerful game development engine, ideal for Hobbyist and Indie developers. Where you can start coding in the easy to learn AppGameKit BASIC or use the libraries in C++ & XCode.

[Amazon Lumberyard](#) is an open source, AAA game engine(based on CryEngine) that gives you the tools you need to create high quality games. Deeply integrated with AWS and Twitch, Amazon Lumberyard includes full source code, allowing you to customize your project at any level.

[Blender](#) is the free and open source 3D creation suite. It supports the entirety of the 3D pipeline—modeling, rigging, animation, simulation, rendering, compositing and motion tracking, video editing and 2D animation pipeline.

[CryEngine](#) is a powerful real-time game development platform created by Crytek.

[GameMaker Studio 2](#) is the latest and greatest incarnation of GameMaker. It has everything you need to take your idea from concept to finished game. With no barriers to entry and powerful functionality, GameMaker Studio 2 is the ultimate 2D development environment.

[Godot](#) is a feature-packed, cross-platform game engine to create 2D and 3D games from a unified interface. It provides a comprehensive set of common tools, so that users can focus on making games without having to reinvent the wheel. Games can be exported in one click to a number of platforms, including the major desktop platforms (Linux, Mac OSX, Windows) as well as mobile (Android, iOS) and web-based (HTML5) platforms.

[Open Graphics Library\(OpenGL\)](#) is an API used across multiple programming languages and platforms for hardware-accelerated rendering of 2D/3D vector graphics currently developed by the [Khronos Group](#).

[Open Computing Language \(OpenCL\)](#) is an open standard for [parallel programming](#) of heterogeneous platforms consisting of CPUs, GPUs, and other hardware accelerators found in supercomputers, cloud servers, personal computers, mobile devices and embedded platforms.

[OpenGL Shading Language\(GLSL\)](#) is a High Level Shading Language based on the C-style language, so it covers most of the features a user would expect with such a language. Such as control structures (for-loops, if-else statements, etc) exist in GLSL, including the switch statement.

[High Level Shading Language\(HLSL\)](#) is the High Level Shading Language for DirectX. Using HLSL, the user can create C-like programmable shaders for the Direct3D pipeline. HLSL was first created with DirectX 9 to set up the programmable 3D pipeline.

[DirectX 12 Ultimate](#) is an API(for high performance 2D & 3D graphics) from Microsoft. DirectX 12 Ultimate brings support for ray tracing, mesh shaders, variable rate shading, and sampler feedback. Available in Windows 2004 version(May 2020 Update).

[Vulkan](#) is a modern cross-platform graphics and compute API that provides high-efficiency, cross-platform access to modern GPUs used in a wide variety of devices from PCs and consoles to mobile phones and embedded platforms. Vulkan is currently in development by the Khronos consortium.

[Metal](#) is a low-level GPU programming framework used for rendering 2D and 3D graphics on Apple platforms such as iOS, iPadOS, macOS, watchOS and tvOS.

[MoltenVK](#) is an implementation of Vulkan running on iOS and macOS using Apple's [Metal](#) graphics framework.

[MoltenGL](#) is an implementation of the OpenGL ES 2.0 API that runs on Apple's [Metal](#) graphics framework.

[AMD Radeon ProRender](#) is a powerful physically-based rendering engine that enables creative professionals to produce stunningly photorealistic images on virtually any GPU, any CPU, and any OS in over a dozen leading digital content creation and CAD applications.

[NVIDIA Omniverse](#) is a powerful, multi-GPU, real-time simulation and collaboration platform for 3D production pipelines based on Pixar's Universal Scene Description and NVIDIA RTX.

[LibGDX](#) is a cross-platform Java game development framework based on OpenGL (ES) that works on Windows, Linux, Mac OS X, Android, your WebGL enabled browser and iOS.

[cocos2d-x](#) is a multi-platform framework for building 2d games, interactive books, demos and other graphical applications. It is based on cocos2d-iphone, but instead of using Objective-C, it uses C++. It works on iOS, Android, macOS, Windows and Linux.

[MonoGame](#) is a framework for creating powerful cross-platform games. The spiritual successor to XNA with thousands of titles shipped across desktop, mobile, and console platforms. MonoGame is a fully managed .NET open source game framework without any black boxes.

[Three.js](#) is a cross-browser JavaScript library and application programming interface used to create and display animated 3D computer graphics in a web browser using WebGL.

[Superpowers](#) is a downloadable HTML5 app for real-time collaborative projects . You can use it solo like a regular offline game maker, or setup a password and let friends join in on your project through their Web browser.

[URHO3D](#) is a free lightweight, cross-platform 2D and 3D game engine implemented in C++ and released under the MIT license. Greatly inspired by OGRE and Horde3D.

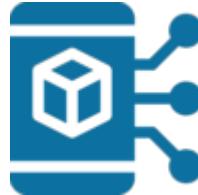
[Vivox](#) is a voice & text chat platform that's trusted by the world's biggest gaming brands and titles such as Fortnite, PUBG, League of Legends, and Rainbow Six Siege.

[HGIG](#) is a volunteer group of companies from the game and TV display industries that meet to specify and make available for the public guidelines to improve consumer gaming experiences in HDR.

[GameBlocks](#) is a Server Side Anti-Cheat & Middleware software.

55. Blockchain

[Back to the Top](#)



Blockchain Guide

A guide covering Blockchain including the applications and tools that will make you a better and more efficient Blockchain developer.

[Blockchain](#) is a distributed database meaning that the storage devices used for the database are not all connected to a common processor. It maintains a growing decentralized digital list of ordered records, called blocks. Each block has a timestamp and a link to a previous block. By the system being decentralization it allows for complete transparency in all shared information.

[IBM Blockchain Essentials](#)

[Blockchain Courses by Coursera](#)

[Blockchain Courses by edX](#)

[Blockchain Certifications](#)

Tools

[Hyperledger](#) is a new Collaborative Project at The Linux Foundation. The technical community is just getting started and will be adding code to the repository in the coming weeks. Check [hyperledger.org](#) for more information about joining the mailing lists and participating in the conversations.

[Hyperledger Fabric](#) is an enterprise-grade permissioned distributed ledger framework for developing solutions and applications. Its modular and versatile design satisfies a broad range of industry use cases. It offers a unique approach to consensus that enables performance at scale while preserving privacy.

[Geth](#) is an implementation of an [Ethereum](#) node in the Go programming language.

[Solidity](#) is a purposefully slimmed down, loosely-typed language with a syntax very similar to ECMAScript (Javascript) that is used for the creation of smart contracts on the Ethereum blockchain.

[GanacheCLI \(NodeJS package\)](#) is a fast and customizable blockchain emulator. It simulates the Ethereum network on a single computer and allows you to make calls to the blockchain without any of the hassles of

running a real Ethereum node.

[Truffle](#) is a development environment, testing framework and asset pipeline for Ethereum, aiming to make life as an Ethereum developer easier.

[Ethers.js](#) is a library which was made for and is used by ethers.io. It is designed to make it easier to write client-side JavaScript based wallets, keeping the private key on the owner's machine at all times.

[OpenEthereum](#) is a fast and feature-rich multi-network Ethereum client.

[Quorum](#) is an Ethereum-based distributed ledger protocol with transaction/contract privacy and new consensus mechanisms.

[Diesel](#) is a safe, extensible ORM and Query Builder for [Rust](#).

[BlockShell](#) is a minimal command line utility for learning Blockchain technical concepts like chaining & mining.

[Tendermint Core](#) is [Byzantine Fault Tolerant \(BFT\)](#) middleware that takes a state transition machine written in any programming language and securely replicates it on many machines.

[inDEX](#) is a comprehensive list of decentralized exchanges (DEX) of cryptocurrencies, tokens, derivatives and futures, and their protocols.

[Squeezer](#) is a platform that empowers new-entry blockchain developers to build serverless dApps easily.

56. Working Remote

[Back to the Top](#)



Working Remote Guide

This guide will go over the tools and techniques needed to make you more efficient working from home.

Resources

[Remote Working Solutions from Microsoft](#)

[How stay productive while working remotely with Microsoft Teams](#)

[Distance & Remote Learning from Microsoft Education](#)

[Remote Working: Setting Yourself and Your Teams Up for Success from LinkedIn Learning](#)

[Productivity Masterclass - Principles and Tools to Boost Your Productivity by Ali Abdaal](#)

[Best Practices for Working Remotely from Udemy](#)

[Learning new remote work strategies courses from Coursera](#)

[GitLab's Guide to Going All-Remote](#)

[Best Practices for Working Remotely from Cisco](#)

[The benefits of remote work for both employees and managers from WeWork](#)

[Working Remotely by Harvard Business Publishing](#)

[Why the 2020s will be the Remote Work Decade by Chris Herd.](#)

Tools

[Zoom](#) is the leader in modern enterprise video communications, with an easy, reliable cloud platform for video and audio conferencing, chat, and webinars across mobile, desktop, and room systems. Zoom Rooms is the original software-based conference room solution used around the world in board, conference, huddle, and training rooms, as well as executive offices and classrooms.

[Microsoft 365](#) is an integrated cloud platform that delivers industry-leading productivity apps like [Microsoft Teams](#), [Skype](#), Word, Excel, and PowerPoint, along with intelligent cloud services and world-class security.

[Google Workspace](#) is a suite of cloud computing, productivity and collaboration tools, software and products developed by Google such as Gmail, Docs, Drive, Calendar, Meet(video chat) and more for business.

[BlueJeans](#) is the world's leader in cloud video conferencing. Easily and securely hold live video calls, webinars, conference calls, and online meetings. Has been acquired by Verizon as of May 2020.

[Jitsi meet](#) is a fully encrypted, 100% open source video conferencing solution that you can use all day, every day, for free — with no account needed.

[Slack](#) is a team communication and collaboration hub all rolled into one place so your team can get more work done, whether you belong to a large enterprise or a small business.

[Mattermost](#) is an open source Slack alternative. It's written in Golang and React and runs as a single Linux binary with MySQL or PostgreSQL. Use the features you like (file sharing, real-time group chat and webhooks—to name a few) and access the source code. Mattermost mostly markets itself as an open-source alternative to Slack and Microsoft Teams.

[Discord](#) is an instant messaging and VoIP application platform designed for creating communities ranging from gamers, education, and businesses. Easily create/enter chat rooms, initiate video calls, and create multiple groups for messaging friends. Discord specializes in text, image, video and audio communication between users in a chat channel.

[Twitch](#) is the world's leading video and community platform that brings large groups of people together. Whether you're there for games, live music, educational tutorials, and other creative pieces of content.

[Amazon Interactive Video Service \(Amazon IVS\)](#) is a managed live streaming solution that is quick and easy to set up, and ideal for creating interactive video experiences. Send your live streams to Amazon IVS using streaming software and the service does everything you need to make low-latency(under 5 seconds) live video available to any viewer around the world, letting you focus on building interactive experiences alongside the live video.

[Synology Drive](#) is a data management and collaboration tool that synchronizes files across various platforms, fulfilling your office online editing and seamless granular sharing needs with Google Drive, Dropbox, and Box right on your Synology NAS.

[Syncthing](#) is an open source continuous file synchronization program. It synchronizes files between two or more computers in real time, safely protecting your data from prying eyes.

[Citrix Virtual Apps and Desktops](#) is a virtualization solution for application and desktop delivery to any device, over any network, while enhancing data security, reducing costs, and empowering productivity.

[TeamViewer](#) is a feature packed remote access software that enables people to share their screens with other devices. The remote management suite includes chat messaging, audio calls, and video communication during your sessions.

57. Audio & Video Editing

[Back to the Top](#)



Audio and Video Editing Guide

A guide covering applications, tools, and techniques that will make you a better and more efficient Audio/Video Editor.

Resources

[10 Video Production Tools That Make the Difference Between a Good Video and a Bad One by Dann Albright](#)

[Online Audio Editing Classes from Skillshare](#)

[Online Video Editing Classes Skillshare](#)

[Video Editing Courses from Udemy](#)

[Audio Editing Courses from udemy](#)

[Video Editing Courses from Coursera](#)

[The Basics of Audio Editing from Coursera](#)

[Audacity Podcast Production Course: Audio Editing for Podcasters](#)

[Video Editing Online Training Courses from LinkedIn Learning](#)

[Audio Editing Online Training Courses from LinkedIn Learning](#)

Tools and Equipment

[H.264\(AVC\)](#) is a video compression standard based on block-oriented and motion-compensated integer-DCT coding that defines multiple profiles (tools) and levels (max bitrates and resolutions) with support up to 8K.

[H.265\(HEVC\)](#) is a video compression standard that is the successor to H.264(AVC). It offers a 25% to 50% better data compression at the same level of video quality, or improved video quality at the same bit-rate.

[FFmpeg](#) is a leading multimedia framework that can decode, encode, transcode, mux, demux, stream, filter and play pretty much anything that humans and machines have created. It supports the most obscure ancient formats up to the cutting edge ones on multiple platforms such as Windows, macOS, and Linux.

[Intel® Quick Sync Video](#) is a dedicated media processing capabilities of Intel® Graphics Technology to decode and encode fast, enabling the processor to complete other tasks and improving system responsiveness

[HandBrake](#) is a tool for transcoding video from almost any format with a selection of widely supported codecs. It is supported on Window, macOS, and Linux.

[HTTP Live Streaming \(HLS\)](#) is a communications protocol developed by Apple that sends live and on-demand audio and video to iPhone, iPad, Mac, Apple Watch, Apple TV, and PC.

[Dynamic Adaptive Streaming over HTTP \(DASH\)](#) is an adaptive streaming protocol that allows for a video stream to switch between bit rates on the basis of network performance, in order to keep a video playing.

[OpenMAX™](#) is a cross-platform API that provides comprehensive streaming media codec and application portability by enabling accelerated multimedia components to be developed, integrated and programmed across multiple operating systems and silicon platforms.

[Apple ProRes](#) is a codec technology developed by Apple for high-quality & high-performance editing in Final Cut Pro that supports up to 8K.

[Premiere Pro](#) is the industry-leading video editing software for film, TV, and the web. Creative tools, integration with other apps and services, and the power of Adobe Sensei help you craft footage into polished films and videos. With [Premiere Rush](#) you can create and edit new projects from any device.

[Final Cut Pro](#) is Apple's professional grade video editing software.

[Kdenlive](#) is an open source video editing tool that supports unlimited multimedia files. It's based on the MLT Framework, KDE and Qt. People who are looking for a very versatile video editing tool that comes packed with features. The latest 20.08 release is out with nifty features like Interface Layouts, Multiple Audio Stream support, Cached data management and Zoombars in the Clip Monitor and Effects Panel but one may argue that the highlights of this release are stability and interface improvements.

[OpenShot](#) is an open-source video editing tool that's designed for users new in the editing environment. It has simple features such as a simple drag-and-drop function, it provides an easy-to-use and quick-to-learn user interface. The powerful video editor offers tons of efficient ways to cut and trim down your videos. You can freely utilize the unlimited tracks, video effects engine, title editor, 3D animations, slow motion, and time effects. It supports commonly used video codecs that are supported by FFmpeg like WebM (VP9), AVCHD (libx264), HEVC (libx265) and audio codecs like mp3 (libmp3lame) and aac (libfaac). The program can render MPEG4, ogv, Blu-ray and DVD video, and Full HD videos for uploading to the internet video websites like YouTube.

[DaVinci Resolve 16](#) is the world's only solution that combines professional 8K editing, color correction, visual effects and audio post production all in one software tool! You can instantly move between editing, color, effects, and audio with a single click. DaVinci Resolve Studio is also the only solution designed for multi user collaboration so editors, assistants, colorists, VFX artists and sound designers can all work live on the same project at the same time.

[Blender](#) comes with a built-in video sequence editor allows you to perform basic actions like video cuts and splicing, as well as more complex tasks like video masking or color grading. The Video Editor includes: Live preview, luma waveform, chroma vectorscope and histogram displays. Audio mixing, syncing, scrubbing and waveform visualization.

[Lightworks](#) is a non-linear video editing application for editing and mastering digital video used by the film industry. Its professional edition has been used for box office hits, such as Shutter Island, Pulp Fiction, and Mission Impossible. Intimidating user interface. Like professional video editors, such as Adobe Premiere Pro, Lightworks is rather complicated to use for new users.

[Shotcut](#) is an open source multi-platform video editor. You can perform various actions such as video editing (including 4K video quality), add effects, create new movies, import most image files formats, export to almost any file format and much more.

[Olive](#) is a free non-linear video editor aiming to provide a fully-featured alternative to high-end professional video editing software.

[OBS \(Open Broadcaster Software\)](#) is free and open source software for video recording and live streaming. Stream to Twitch, YouTube and many other providers or record your own videos with high quality H264 / AAC encoding.

[REAPER](#) is a complete digital audio production application for computers, offering a full multitrack audio and MIDI recording, editing, processing, mixing and mastering toolset. REAPER supports a vast range of hardware, digital formats and plugins, and can be comprehensively extended, scripted and modified.

[JACK Audio Connection Kit AKA JACK](#) is a professional sound server daemon that provides real-time, low-latency connections for both audio and MIDI data between applications that implement its API. JACK can be configured to send audio data over a network to a "master" machine, which then outputs the audio to a physical device. This can be useful to mix audio from a number of "slave" computers without requiring additional cables or hardware mixers, and keeping the audio path digital for as long as possible.

[Logic Pro](#) is a digital audio workstation (DAW) and MIDI sequencer software application for macOS.

[Bitwig Studio](#) is a digital audio workstation that has linear and non-linear workflows for sound design, recording, live performance, and more. Along with 90+ instruments, effects, and other creative tools. It is

supported Windows, macOS, and Linux.

[PipeWire](#) is a server and user space API to deal with multimedia pipelines. It provides a low-latency, graph based processing engine on top of audio and video devices that can be used to support the use cases currently handled by both pulseaudio and JACK. PipeWire was designed with a powerful security model that makes interacting with audio and video devices from containerized applications easy. Nodes in the graph can be implemented as separate processes, communicating with sockets and exchanging multimedia content using fd passing.

[LMMS](#) is an open source digital audio workstation application program. When LMMS is paired with appropriate computer hardware, it allows music to be produced by arranging samples, synthesizing sounds, playing on a MIDI keyboard, and combining the features of trackers and sequencers. Developed by Paul Giblock and Tobias Junghans, this program stands for "Linux MultiMedia Studio" and supports handy plugins that enable it to work on different operating systems.

[Ardour](#) is an open source, collaborative effort of a worldwide team including musicians, programmers, and professional recording engineers. Development is transparent — anyone can watch our work as it happens. Like a good piece of vintage hardware, you can open the box and look inside.

[Audacity](#) is an easy-to-use, multi-track audio editor and recorder for Windows, Mac OS X, GNU/Linux and other operating systems. Developed by a group of volunteers as open source and offered free of charge. Amazing support community.

[Glimpse](#) is a cross-platform raster graphics editor based on the GNU Image Manipulation Program available for Linux, MacOS, and Windows. A great tool for making YouTube video thumbnails.

58. Podcasting

[Back to the Top](#)



Podcasting Guide

A guide covering applications, tools, and techniques that will make you a better and more efficient Podcaster.

Resources and Equipment

[Online Podcasting Classes from Skillshare](#)

[Audacity Podcast Production Course: Audio Editing for Podcasters](#)

[Best Podcast Microphones For Great Audio \(All Budgets\) by Ross Winn](#)

[The Best Podcast Equipment for Professional Podcast Studio Setup](#)

Live Streaming and Elgato Game Capture HD

[Open Broadcaster Software \(OBS\) streaming with Elgato Game Capture HD](#)

[What is Amazon Interactive Video Service?](#)

Tools

[Fireside](#) provides every podcast with a free SSL ([https](https://https://)) certificate for both subdomains and your own custom domain name, ensuring your website and all of its content is delivered to your audience securely.

[Anchor](#) is an all-in-one platform where you can create, distribute, and monetize your podcast from any device, for free.

[Wavve](#) is a tool to create shareable audio clips for social media.

[Amazon Interactive Video Service \(Amazon IVS\)](#) is a managed live streaming solution that is quick and easy to set up, and ideal for creating interactive video experiences. Send your live streams to Amazon IVS using streaming software and the service does everything you need to make low-latency(under 5 seconds) live video available to any viewer around the world, letting you focus on building interactive experiences alongside the live video.

[Streamlabs](#) is the most popular cloud-based streaming platform for Twitch(70% usage), YouTube, Mixer and Facebook. Easily grow your coomunity with Streamlabs Open Broadcast Software (OBS), alerts, 1000+ overlays, analytics, chatbot, tipping, merch and more. Available on Windows and MacOS.

[OBS \(Open Broadcaster Software\)](#) is free and open source software for video recording and live streaming. Stream to Twitch, YouTube and many other providers or record your own videos with high quality H264 / AAC encoding.

[Opencast](#) is a flexible, scalable and reliable video capture, management, and distribution system for academic institutions. Opencast is built by a growing community of developers from leading universities and organizations worldwide. Also checkout [Opencast Studio](#).

[audio-visualizer-python](#) is a little GUI tool which creates an audio visualization video from an input audio. You can also give it a background image and set a title text.

[Bitwig Studio](#) is a digital audio workstation that has linear and non-linear workflows for sound design, recording, live performance, and more. Along with 90+ instruments, effects, and other creative tools. It is supported Windows, macOS, and Linux.

[LBRY](#) is a blockchain-based digital content protocol and community-controlled content wonderland containing a digital library with millions of pieces of content, such as videos, music, games.

[SoundCaster](#) connects your digital audio devices and broadcasts multi-channel audio to your phones and tablets.

[Soundboard Studio](#) is a fast and versatile cartwall app for iOS. Perfect for for pads, music beds, voiceovers, sound effects, soundbites and much more.

[UniversalSoundboard](#) is a customizable soundboard for your Windows 10 device. Add your sounds or music and sort them in categories. You can also add images to your favourite sounds.

[CastersoundBoard](#) is an open source soundboard for hot-keying and playing back sounds for podcasting.

[PipeWire](#) is a server and user space API to deal with multimedia pipelines. It provides a low-latency, graph based processing engine on top of audio and video devices that can be used to support the use cases currently handled by both pulseaudio and JACK. PipeWire was designed with a powerful security model that makes interacting with audio and video devices from containerized applications easy. Nodes in the graph can be implemented as separate processes, communicating with sockets and exchanging multimedia content using fd passing.

[REAPER](#) is a complete digital audio production application for computers, offering a full multitrack audio and MIDI recording, editing, processing, mixing and mastering toolset. REAPER supports a vast range of hardware, digital formats and plugins, and can be comprehensively extended, scripted and modified.

[JACK Audio Connection Kit AKA JACK](#) is a professional sound server daemon that provides real-time, low-latency connections for both audio and MIDI data between applications that implement its API. JACK can be configured to send audio data over a network to a "master" machine, which then outputs the audio to a physical device. This can be useful to mix audio from a number of "slave" computers without requiring additional cables or hardware mixers, and keeping the audio path digital for as long as possible.

[Ardour](#) is an open source, collaborative effort of a worldwide team including musicians, programmers, and professional recording engineers. Development is transparent — anyone can watch our work as it happens. Like a good piece of vintage hardware, you can open the box and look inside.

[LMMS](#) is an open source digital audio workstation application program. When LMMS is paired with appropriate computer hardware, it allows music to be produced by arranging samples, synthesizing sounds, playing on a MIDI keyboard, and combining the features of trackers and sequencers. Developed by Paul Giblock and Tobias Junghans, this program stands for "Linux MultiMedia Studio" and supports handy plugins that enable it to work on different operating systems.

[Audacity](#) is an easy-to-use, multi-track audio editor and recorder for Windows, Mac OS X, GNU/Linux and other operating systems. Developed by a group of volunteers as open source and offered free of charge. Amazing support community.

59. Agile Development

[Back to the Top](#)



Agile Development Guide

A guide that covers Agile Methodologies, Disciplines, and Certifications that make you better and more efficient at managing development Teams.

Resources

[What is Agile Development? - Azure DevOps](#)

[What is Agile? - Atlassian](#)

[What is Agile Software Development? - Agile Alliance](#)

[Agile Development from Coursera](#)

[Learn Agile project management from Udemy](#)

[Learn Agile with Online Courses from edX](#)

[Online Agile Classes from Skillshare](#)

[Agile Programming Training & Certification from Learning Tree](#)

Project Management Professional (PMP) certification tells employers, peers, and the world you know what you're doing in project management.

Program Management Professional (PgMP) certification is a visible sign of your advanced experience and skill and gives you a distinct advantage in employment and promotion.

Portfolio Management Professional (PfMP) certification signifies your advanced competency in the coordinated management of one or more portfolios to achieve strategic objectives.

PMI Agile Certified Practitioner (PMI-ACP) certification formally recognizes your knowledge of agile principles and your skill with agile techniques. It shows your peers, employers, and stakeholders that your agile knowledge runs deep.

PMI Risk Management Professional (PMI-RMP) certification highlights your ability to identify and assess project risks, mitigate threats and capitalize on opportunities. In this capacity, you enhance/protect the needs of your organization.

Disciplined Agilist (DA) certification indicates to colleagues and employers that you have taken some initial steps to learn about Disciplined Agile.

Certified Disciplined Agilist (CDA) certification indicates to colleagues and employers that you have fundamental knowledge about Disciplined Agile strategies and that you are eager to learn and increase your skills and abilities as a software professional.

Certified Disciplined Agile Practitioner (CDAP) certification indicates that you are experienced at agile solution delivery and are on your way to becoming a generalizing specialist. By having the potential to be a "junior coach" under the guidance of a senior coach.

Disciplined Agile Lean Scrum Master (DALSM) certification provides tools and strategies to help you lead an agile team, influence people, remove impediments and to tailor your team's way of working (WoW) so it matches and reflects the unique situation of your unique team.

Certified Disciplined Agile Coach (CDAC) certification indicates that you are a trusted expert with significant proficiency with Disciplined Agile. You can coach other people in disciplined agile strategies and advise organizations in the adoption and tailoring of Disciplined Agile.

[Certified Disciplined Agile Instructor \(CDAI\)](#) certification indicates being a trusted expert with proficiency in Disciplined Agile.

Agile Methodologies & Disciplines

[Scrum](#) is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value.

[Lean](#) is an agile methodology that eliminates waste through such practices as selecting only the truly valuable features for a system, prioritizing those selected, and delivering them in small batches. It emphasizes the speed and efficiency of development workflow, and relies on rapid and reliable feedback between programmers and customers. Lean uses the idea of work product being "pulled" via customer request. It focuses decision-making authority and ability on individuals and small teams, since research shows this to be faster and more efficient than hierarchical flow of control. Lean also concentrates on the efficiency of the use of team resources, trying to ensure that everyone is productive as much of the time as possible. It concentrates on concurrent work and the fewest possible intra-team workflow dependencies. Lean also strongly recommends that automated unit tests be written at the same time the code is written.

[Kanban](#) is a scheduling system for lean manufacturing and just-in-time manufacturing. Like Scrum, Kanban is a process designed to help teams work together more effectively.

[Kanban board](#) is an agile project management tool designed to help visualize work, limit work-in-progress, and maximize efficiency (or flow). Kanban boards use cards, columns, and continuous improvement to help technology and service teams commit to the right amount of work, and get it done.

[Dynamic System Development Model, \(DSDM\)](#) is an agile framework that addresses the entire project lifecycle and its impact on the business. Like the broader agile philosophy, DSDM is an iterative approach to software development, and this framework explicitly states "any project must be aligned to clearly defined strategic goals and focus upon early delivery of real benefits to the business." The framework is built on four principles: feasibility and business study, functional model and prototype iteration, design and build iteration, and implementation.

[Extreme Programming \(XP\)](#) is an agile software development framework that aims to produce higher quality software, and higher quality of life for the development team. XP is the most specific of the agile frameworks regarding appropriate engineering practices for software development.

[Crystal](#) is an agile framework focusing on individuals and their interactions, as opposed to processes and tools. In other words, this framework is a direct outgrowth of one of the core values articulated in the [Agile Manifesto](#). The Crystal agile framework is built on two core beliefs:

Teams can find ways on their own to improve and optimize their workflows

Every project is unique and always changing, which is why that project's team is best suited to determine how it will tackle the work

[Adaptive software development \(ASD\)](#) is a direct outgrowth of an earlier agile framework, Rapid Application Development (RAD). It aims to enable teams to quickly and effectively adapt to changing

requirements or market needs by evolving their products with lightweight planning and continuous learning.

[Disciplined agile delivery](#) is a people-first, learning-oriented hybrid agile approach to IT solution delivery. It has a risk-value delivery lifecycle, is goal-driven, is enterprise aware, and is scalable.

[Feature-driven development \(FDD\)](#) is an agile framework that, as its name suggests, organizes software development around making progress on features. Features in the FDD context, though, are not necessarily product features in the commonly understood sense.

[Scrumban](#) is an Agile management methodology that combines hybrids of Scrum and Kanban together. It was originally designed as a way to transition from Scrum to Kanban.

[Kanplan](#) is a mixed methodology for practicing agile software development. Like scrumban, it combines features from both scrum and kanban. Kanplan is ideal for teams who want the ability to backlog groom, but don't want to work in sprints.

[RAD\(Rapid Application Development\)](#) is an agile framework focused primarily on rapid prototyping of software products, frequently iterating based on feedback, and continuously releasing updated versions of those products to the market.

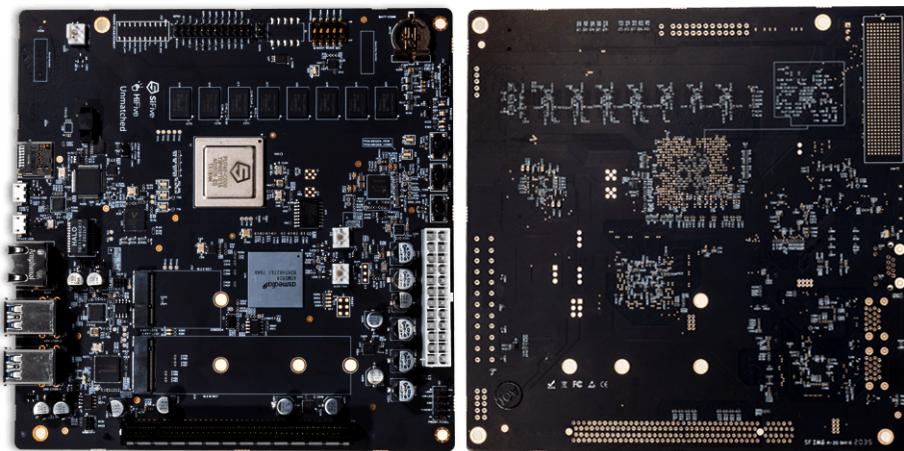
60. RISC-V

[Back to the Top](#)



A guide covering the RISC-V Architecture including the applications, libraries and tools that will make you a better and more efficient developer with the RISC-V ISA.

[Checkout the HiFive Unmatched Developement board](#)



HiFive Unmatched Hardware Specs

SoC: SiFive Freedom U740 SoC

Memory: 16GB DDR4

Flash Memory: 32MB Quad SPI Flash

Removable Storage: MicroSD Card

Networking: Gigabit Ethernet Port

User I/O

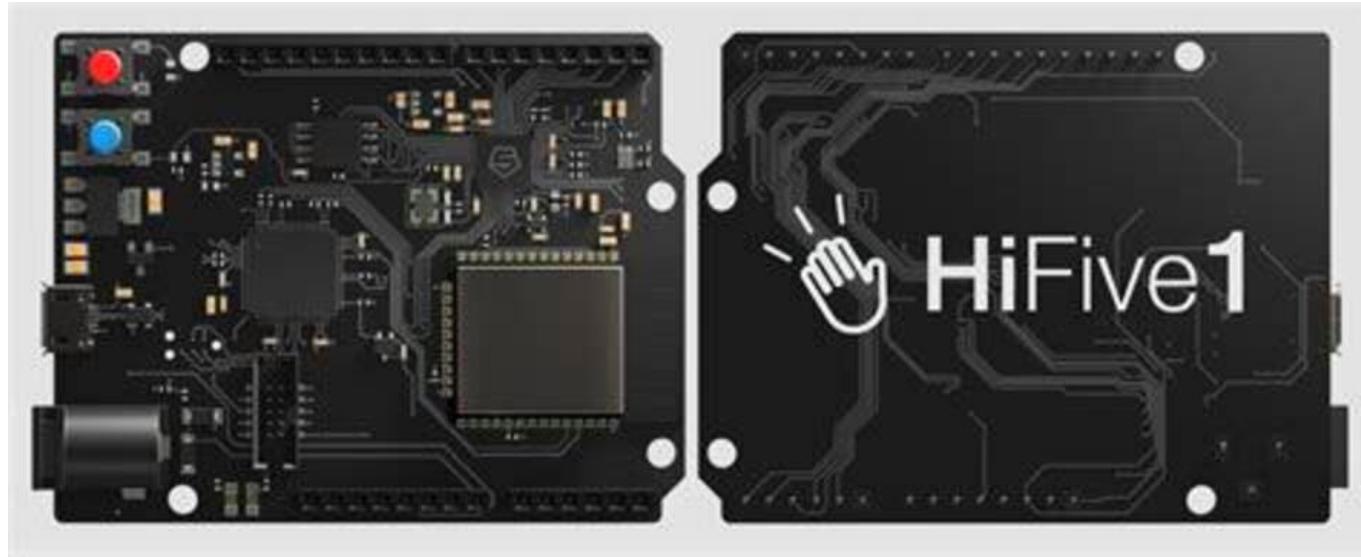
- 4x USB 3.2 Gen 1 Type A Ports (1 Charging Port)
- 1x MicroUSB Console Port

Expansion Capabilities

- x16 PCIe® Gen 3 Expansion Slot (8-lanes Useable)
- M.2 M-Key Slot (PCIe Gen 3 x4) for NVME 2280 SSD Module
- M.2 E-Key Slot (PCIe Gen 3 x1) for Wi-Fi / Bluetooth Module

Board Form Factor: Industry Standard Mini-ITX

[Checkout the HiFive1 Rev B Development board](#)



HiFive1 Rev B Hardware Specs

Microcontroller: [FE310-G002](#)

Operating Voltage: 3.3 V and 1.8 V

Input Voltage: 5 V USB or 7-12 VDC Jack

IO Voltage: 3.3 V

Digital I/O Pins: 19

PWM Pins: 9

SPI Controllers/HW CS Pins: 1/3

UART: 2

I2C: 1

Networking: WiFi/BT (off-chip)

External Interrupt Pins: 19

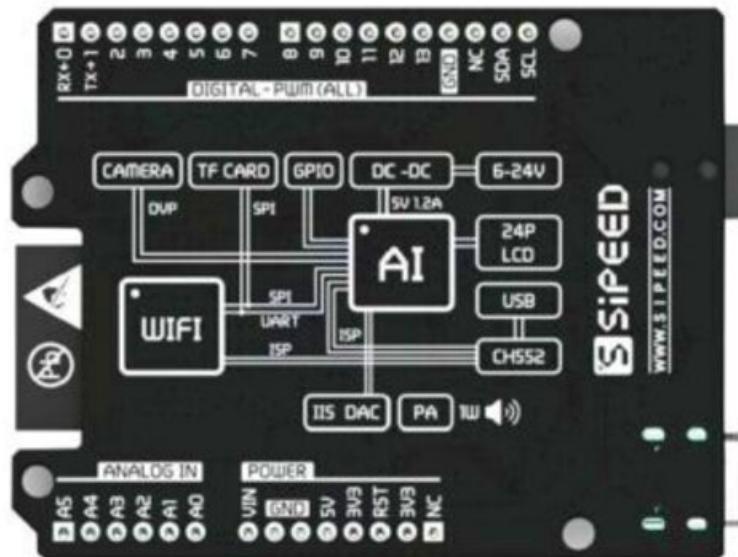
External Wakeup Pins: 1

Flash Memory: 32 Mbit Off-Chip (ISSI SPI Flash)

Host Interface (microUSB): Program, Debug, and Serial Communication

[Checkout the Sipeed Maixduino Kit for RISC-V AI + IoT](#)

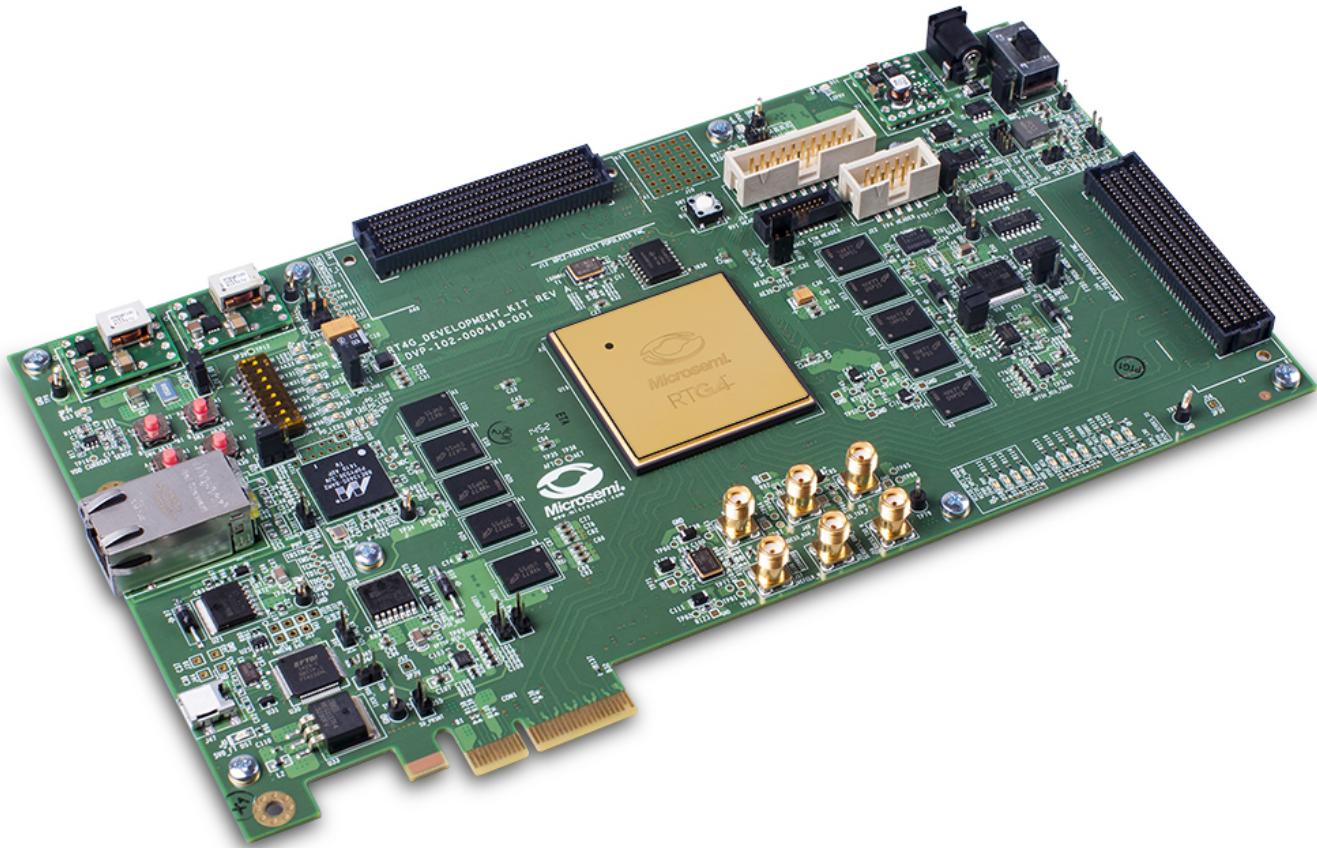
[MAIX AI module](#)



Sipeed Maixduino Kit Hardware Specs

- **CPU:** RISC-V Dual Core 64bit, with FPU; 400MHz neural network processor
- QVGA@60FPS/VGA@30FPS image identification
- Onboard ESP32 module support 2.4G 802.11. b/g/n and Bluetooth 4.2
- [Arduino Uno](#) form factor, Arduino compatible interface
- Onboard omnidirectional I2S digital output MEMS Microphone
- 24P 0.5mm FPC connector for DVP Camera
- 8-bit MCU LCD 24P 0.5mm FPC connector
- Support self-elastic micro SD card holder
- High performance microphone array processor for machine hearing
- Support MaixPy IDE, Arduino IDE, OpenMV IDE, and PlatformIO IDE
- Support Tiny-Yolo, Mobilenet and TensorFlow Lite for deep learning

[Checkout the RTG4 Development Kit board from Microsemi](#)



RTG4 Development Kit Hardware Specs

- Two 1GB DDR3 synchronous dynamic random access memory (SDRAM)
- 2GB SPI flash memory
- PCI Express Gen 1 x1 interface
- PCIe x4 edge connector
- One pair SMA connectors for testing of the full-duplex SERDES channel
- Two FMC connectors with HPC/LPC pinout for expansion
- RJ45 interface for 10/100/1000 Ethernet
- USB micro-AB connector
- Headers for SPI, GPIOs
- FTDI programmer interface to program the external SPI flash
- JTAG programming interface

Resources

[RISC-V Foundation](#) is a non-profit corporation controlled by its 500 members(NVIDIA, Google, Samsung, Raspberry Pi, SiFive, Canonical, and Western Digital) to drive forward the adoption and implementation of the free and open RISC-V instruction set architecture (ISA).

[SiFive](#) is a semiconductor company that produces complete RISC-V processors IP with pre-integrated SiFive Shield, for whole SoC security, and SiFive Insight advanced trace and debug.

[IAR Systems](#) is a company that enables Linux-based Continuous Integration and automated workflows for embedded systems(ARM & RISC-V).

[The RISC-V Instruction Set Manual](#)

[RISC-V Cores and SoC Overview](#)

[Software development kits, Toolchains and Utilities for RISC-V devices](#)

[RISC-V Development Board and RISC-V CPU](#)

[RISC-V CPUs from Microsemi](#)

[RISC-V Community](#)

[RISC-V Main Members Group](#)

[RISC-V Global Forum](#)

[RISC-V Learn Online Courses](#)

[RISC-V Training Partner Program](#)

[SiFive Academy RISC-V Course](#)

[Risc-V Courses on Udemy](#)

[Imagination Technologies Announces the First RISC-V Computer Architecture Course](#)

[RISC-V Design Verification Strategy](#)

Tools

[RISC-V Interpreter](#) is a online Interpreter for RISC-V build by Cornell University.

[Chipyard](#) is an open source framework for agile development of Chisel-based systems-on-chip. It will allow you to leverage the Chisel HDL, Rocket Chip SoC generator, and other [Berkeley](#) projects to produce a RISC-V SoC with everything from MMIO-mapped peripherals to custom accelerators.

[The Eclipse Embedded CDT](#) is a collection of plug-ins for Arm & RISC-V C/C++ developers.

[PlatformIO](#) is a professional collaborative platform for embedded development with no vendor lock-in. It provides support for multiplatforms and frameworks such as IoT, Arduino, CMSIS, ESP-IDF, FreeRTOS, libOpenCM3, mbed OS, Pulp OS, SPL, STM32Cube, Zephyr RTOS, ARM, AVR, Espressif (ESP8266/ESP32), FPGA, MCS-51 (8051), MSP430, Nordic (nRF51/nRF52), NXP i.MX RT, PIC32, RISC-V.

[PlatformIO for VSCode](#) is a plugin that provides support for the PlatformIO IDE on VSCode.

[Ripes](#) is a graphical processor simulator and assembly editor for the RISC-V instruction set architecture(ISA).

[FireSim](#) is an easy-to-use, Scalable, FPGA-accelerated Cycle-accurate Hardware Simulation.

[Tock](#) is an embedded operating system designed for running multiple concurrent, mutually distrustful applications on Cortex-M and RISC-V based embedded platforms. Tock's design centers around protection, both from potentially malicious applications and from device drivers.

[TinyGo](#) is a Go compiler(based on LLVM) intended for use in small places such as microcontrollers, WebAssembly (Wasm), and command-line tools.

[LLVM](#) is a library that has collection of modular/reusable compiler and toolchain components (assemblers, compilers, debuggers, etc.). With these components LLVM can be used as a compiler framework, providing a front-end(parser and lexer) and a back-end (code that converts LLVM's representation to actual machine code).

[Unicorn](#) is a lightweight, multi-platform, multi-architecture CPU emulator framework(ARM, AArch64, M68K, Mips, Sparc, X86) based on [QEMU](#).

[Keystone](#) is a lightweight multi-platform, multi-architecture(Arm, Arm64, Hexagon, Mips, PowerPC, Sparc, SystemZ & X86) assembler framework.

[Reko](#) is a decompiler for machine code binaries.

[Renode](#) is [Antmicro's](#) virtual development framework for multinode embedded networks (both wired and wireless) and is intended to enable a scalable workflow for creating effective, tested and secure IoT systems.

[Jupiter](#) is an open source and education-oriented RISC-V assembler and runtime simulator written in Java.

[RISC-V Rust](#) is a RISC-V processor and peripheral devices emulator project written in Rust and compiled to WebAssembly.

[Diosix](#) is a lightweight, secure, multiprocessor bare-metal hypervisor written in Rust for RISC-V.

[Maixpy](#) is designed to make AIOT programming easier with Micropython running on the embedded AIOT chip [K210](#).

[DarkRISCV](#) is an open souce RISC-V cpu core implemented in Verilog from scratch.

61. Serverless

[Back to the Top](#)



Serverless Guide

A guide covering Serverless architecture including the applications, libraries and tools that will make you a better and more efficient developer with Serverless.

Resources

[Serverless Framework](#) is an open-source web framework written using Node.js, used for building applications on [AWS Lambda](#). The framework gives developers, teams and companies all of the tools they need to build and operate serverless applications, in one simple, powerful and elegant experience.

Serverless Courses & Training

[What is Serverless Computing? by IBM](#)

[What is Serverless? by Red Hat](#)

[What is Serverless computing? by Cloudflare](#)

[Serverless - Deep Dive on AWS \(Amazon Web Services\)](#)

Databases on AWS

[Building Your First Serverless Web App on AWS](#)

[AWS CLI\(Command Line Interface\) Configuration basics](#)

[Getting Started with Serverless on AWS Lambda](#)

[Building a Serverless Data Lake on AWS](#)

[Using AWS Lambda with CloudFront Lambda@Edge](#)

[What is Amazon API Gateway?](#)

[Amazon API Gateway Documentation](#)

[REST API Reference for Amazon API Gateway API](#)

[Creating a REST API as an Amazon S3 proxy in API Gateway](#)

[Azure Serverless on Microsoft Azure](#)

[Red Hat OpenShift Serverless](#)

[Azure Functions Serverless Compute on Microsoft Azure](#)

[Serverless SQL pool with Azure Synapse Analytics](#)

[Serverless Computing Solutions on Google Cloud](#)

[The Complete Multi-Cloud Serverless Course on A Cloud Guru](#)

[Introduction to Serverless on Kubernetes on edX](#)

[AWS Fundamentals: Building Serverless Applications on Coursera](#)

[AWS Lambda and the Serverless Framework Training on Udemy](#)

[Azure Serverless Functions and Logic Apps on Udemy](#)

[Building Serverless Applications in Azure on Pluralsight](#)

[Configuring Serverless Computing in Microsoft Azure on Pluralsight](#)

Tools

[AWS CLI](#) is a unified command line tool to manage your AWS (Amazon Web Services) services.

[AWS Lambda](#) is a serverless compute service that lets you run code without provisioning or managing servers, creating workload-aware cluster scaling logic, maintaining event integrations, or managing runtimes.

[Amazon API Gateway](#) is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale. Using API Gateway, you can create RESTful APIs and WebSocket APIs that enable real-time two-way communication applications. Also, API Gateway supports containerized and serverless workloads, as well as web applications.

[AWS Chalice](#) is a framework for writing serverless apps in python. It allows you to quickly create and deploy applications that use AWS Lambda.

[Amazon Aurora Serverless](#) is an on-demand, auto-scaling configuration for Amazon Aurora (For MySQL & PostgreSQL editions), where the database will automatically start up, shut down, and scale capacity up or down based on the user's application needs.

[AWS Serverless Application Model \(SAM\)](#) is an open-source framework for building serverless applications. It provides shorthand syntax to express functions, APIs, databases, and event source mappings.

[AWS SAM CLI](#) is a CLI tool to build, test, debug, and deploy Serverless applications using AWS SAM.

[AWS Copilot](#) is a command line interface (CLI) that enables customers to quickly launch and easily manage containerized applications on AWS.

[Serverless Express](#) is a library that runs serverless applications and REST APIs using your existing Node.js application framework, on top of AWS Lambda and Amazon API Gateway.

[Laravel Vapor](#) is a serverless deployment platform for Laravel, powered by AWS.

[Kong](#) is a Cloud-Native API Gateway.

[Faasd](#) is a project similar to [OpenFaaS](#), but without the cost and complexity of Kubernetes. It runs on a single host with very modest requirements, making it fast and easy to manage. Under the hood it uses [containerd](#) and [Container Networking Interface \(CNI\)](#) along with the same core OpenFaaS components from the main project.

[Firecracker](#) is an open source virtualization technology that is purpose-built for creating and managing secure, multi-tenant container and function-based services that provide serverless operational models.

[Gloo Edge](#) is a feature-rich, Kubernetes-native ingress controller, and next-generation API gateway built on Envoy.

[BLESS](#) is an SSH Certificate Authority that runs as an AWS Lambda function and is used to sign SSH public keys.

[WebinyJS](#) is a platform for building serverless applications and APIs (Node.js, React, GraphQL).

[Nuclio](#) is a high-Performance Serverless event and data processing platform.

[Claudia](#) is a library that makes it easy to deploy Node.js projects to AWS Lambda and API Gateway. It automates all the error-prone deployment and configuration tasks, and sets everything up the way JavaScript developers expect out of the box.

[Apache OpenWhisk](#) is a serverless functions platform for building cloud applications. It offers a rich programming model for creating serverless APIs from functions, composing functions into serverless workflows, and connecting events to functions using rules and triggers.

[Traefik](#) is an open-source Edge Router that makes publishing your services a fun and easy experience. It receives requests on behalf of your system and finds out which components are responsible for handling them. What sets Traefik apart, besides its many features, is that it automatically discovers the right configuration for your services.

[TiDB](#) is an open-source NewSQL database that supports Hybrid Transactional and Analytical Processing (HTAP) workloads.

[Prisma](#) is open source ORM for Node.js, TypeScript, PostgreSQL, MySQL and SQLite.

[Zappa](#) is a tool for Serverless Python.

[Dapr](#) is a portable, event-driven, runtime for building distributed applications across cloud and edge.

[Pulumi](#) is the easiest way to create and deploy cloud software that use containers, serverless functions, hosted services, and infrastructure, on any cloud.

[Fission](#) is a fast serverless framework for Kubernetes with a focus on developer productivity and high performance.

[Kubeless](#) is a Kubernetes-native serverless framework that lets you deploy small bits of code without having to worry about the underlying infrastructure plumbing. It leverages Kubernetes resources to provide auto-scaling, API routing, monitoring, troubleshooting and more.

[Knative Serving](#) is a project that builds on Kubernetes to support deploying and serving of applications and functions as serverless containers.

[KEDA](#) is a Kubernetes-based Event Driven Autoscaling component. It provides event driven scale for any container running in Kubernetes.

[StreamAlert](#) is a serverless, realtime data analysis framework which empowers you to ingest, analyze, and alert on data from any environment, using datasources and alerting logic the user defines.

[Serverless Next.js Component](#) is a zero configuration Next.js 9.0 serverless component for [AWS Lambda@Edge](#).

[IronFunctions](#) is an open source serverless platform.

[Squeezee](#) is a framework build serverless dApps.

[Embark](#) is a framework that allows you to easily develop and deploy Decentralized Applications (DApps).

[Midway Serverless](#) is a serverless framework used to build Node.js cloud functions. Helps you significantly reduce maintenance costs and focus more on product development in the Cloud-Native environment.

[Micronaut](#) is a modern, JVM-based, full stack Java framework designed for building modular, easily testable JVM applications with support for Java, Kotlin and the Groovy language.

[Cube.js](#) is an open-source analytical API platform. It is primarily used to build internal business intelligence tools or add customer-facing analytics to existing applications.

62. XML

[Back to the Top](#)



XML Guide

A guide covering XML (Extensible Markup Language) including the applications, libraries and tools that will make you a better and more efficient developer with XML.

Resources

[XML\(Extensible Markup Language\)](#) is a markup language that is similar to HTML but without predefined tags. Instead, the user define their own tags designed specifically for their needs.

[XAML](#) is a declarative XML-based markup language that is applied to the .NET Core programming model, which XAML simplifies creating a UI for a [.NET Core app](#).

[Using XML editor in Visual Studio](#)

[XMLmind XML Editor Tutorial](#)

[XMLReader for Android Developers](#)

[Adding app configuration policies for managed iOS/iPadOS devices](#)

[XML Processing Modules for Python 3](#)

[PHP: XML Parser Manual](#)

[Introduction to XML on W3Schools](#)

[Introduction to XML by IBM](#)

[XML Fundamentals Training Program from Oracle University](#)

[XML Online Training Courses on Udemy](#)

[XML Schema \(XSD\) Crash Course for Beginners on Udemy](#)

[XML Online Training Courses on Coursera](#)

[XML Online Training Courses on LinkedIn Learning](#)

[Introduction to Microsoft Office Open XML on Pluralsight](#)

[XML Fundamentals in C# on Pluralsight](#)

[Working with XML in Java Using JAXB](#)

[Querying JSON, XML, and Temporal Data with T-SQL](#)

[XML Training Courses on NobleProg](#)

Tools

[Visual Studio](#) is an integrated development environment (IDE) from Microsoft; which is a feature-rich application that can be used for many aspects of software development. Visual Studio makes it easy to edit, debug, build, and publish your app. By using Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, and Windows Store.

[MSBuild](#) is the build platform for .NET and Visual Studio. MSBuild, provides an XML schema for a project file that controls how the build platform processes and builds software. Visual Studio uses MSBuild to perform team builds through Azure DevOps Server, but MSBuild can run without Visual Studio.

[WebStorm](#) is a professional IDE for JavaScript(including support for both HTML and CSS) developed by JetBrains. WebStorm comes with intelligent code completion, on-the-fly error detection, powerful navigation and refactoring for JavaScript, TypeScript, stylesheet languages, and all the most popular frameworks([Angular](#), [React](#), [Vue.js](#), [Ionic](#), [Apache Cordova](#), [React Native](#), [Node.js](#), [Meteor](#), and [Electron](#)).

[Uno Platform](#) is a Universal Windows Platform Bridge that allows UWP-based code (C# and XAML) to run on iOS, Android, macOS, WebAssembly, Linux and Windows. It provides the full definitions of the UWP [Windows 10 2004 \(19041\)](#), and the implementation of a growing number of parts of the UWP API, such as Windows.UI.Xaml, to enable UWP and WinUI applications to run on these platforms.

[Android Studio](#) is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development. Available on Windows, macOS, Linux, Chrome OS.

[Oxygen XML Editor](#) is a cross platform tool setting the standard in XML editing. The advanced XML editor provides the most intuitive tools for XML editing, authoring, and development. It is available on Windows, macOS, and Linux.

[XMLmind XML Editor](#) is a strictly validating XML editor. It has Extensive DITA 1.3 support, and includes lightweight DITA, XDITA, HDITA, MDITA.

[Refit](#) is the automatic type-safe REST library for .NET Core,.NET and Xamarin.

[Pugixml](#) is a C++ XML processing library, which consists of a DOM-like interface with rich traversal/modification capabilities, an extremely fast XML parser which constructs the DOM tree from an

XML file/buffer, and an XPath 1.0 implementation for complex data-driven tree queries.

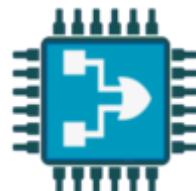
[Serializer](#) is a component that makes it easier to handle serializing data structures, including object graphs, into array structures or other formats like XML and JSON. It can also handle deserializing XML and JSON back to object graphs.

[Erlsom](#) is an XML parser for Erlang.

[REST-assured](#) is a Java DSL for easy testing of REST services.

63. Verilog/SystemVerilog

[Back to the Top](#)



Verilog/SystemVerilog Guide

A guide covering Verilog & SystemVerilog including the applications, libraries and tools that will make you a better and more efficient developer by having a better understanding of how hardware works on the lowest level.

Resources

[Verilog](#) is a Hardware Description Language(HDL) used to design and document electronic systems. Verilog HDL allows designers to design at various levels of abstraction.

[SystemVerilog](#) is an extension of Verilog with many of the verification features that allow engineers to verify the design using complex testbench structures and random stimuli in simulation.

[Verilog Book Shelf](#)

[Verilog HDL Basics training from Intel](#)

[SystemVerilog for Design and Verification](#)

[Verilog HDL Programming Courses on Udemy](#)

[Top Verilog Programming Courses on Coursera](#)

[Verilog course for Engineers on TechnobYTE](#)

[Verilog Tutorials and Courses on hackr.io](#)

[Designing With Verilog Certification from the Xilinx Learning Center](#)

[Learning Verilog for FPGA Development on LinkedIn Learning](#)

[SystemVerilog tutorial on ChipVerify](#)

Tools

[Apio](#) is a multiplatform toolbox, with static pre-built packages, project configuration tools and easy command interface to verify, synthesize, simulate and upload your verilog designs.

[IceStorm](#) is a project that aims at documenting the bitstream format of Lattice iCE40 FPGAs and providing simple tools for analyzing and creating bitstream files.

[Icestudio](#) is a visual editor for open FPGA boards. Built on top of the Icestorm project using Apio.

[EDA Playground](#) is a online code for programming your Verilog projects.

[PlatformIO](#) is a professional collaborative platform for embedded development with no vendor lock-in. It provides support for multiplatforms and frameworks such as IoT, Arduino, CMSIS, ESP-IDF, FreeRTOS, libOpenCM3, mbed OS, Pulp OS, SPL, STM32Cube, Zephyr RTOS, ARM, AVR, Espressif (ESP8266/ESP32), FPGA, MCS-51 (8051), MSP430, Nordic (nRF51/nRF52), NXP i.MX RT, PIC32, RISC-V.

[PlatformIO for VSCode](#) is a plugin that provides support for the PlatformIO IDE on VSCode.

[Chisel](#) is a hardware design language that facilitates advanced circuit generation and design reuse for both ASIC and FPGA digital logic designs. Chisel adds hardware construction primitives to the [Scala](#) programming language, providing designers with the power of a modern programming language to write complex, parameterizable circuit generators that produce synthesizable Verilog.

[Clash compiler](#) is a functional hardware description language that borrows both its syntax and semantics from the functional programming language Haskell. The Clash compiler transforms these high-level descriptions to low-level synthesizable VHDL, Verilog, or SystemVerilog.

[Verilator](#) is an open-source SystemVerilog simulator and lint system.

[Verilog to Routing\(VTR\)](#) is a collaborative project to provide a open-source framework for conducting FPGA architecture and CAD Research & Development. The VTR design flow takes as input a Verilog description of a digital circuit, and a description of the target FPGA architecture.

[Cascade](#) is a Just-In-Time Compiler for Verilog from VMware Research. Cascade executes code immediately in a software simulator, and performs compilation in the background. When compilation is finished, the code is moved into hardware, and from the user's perspective it simply gets faster over time.

[OpenTimer](#) is a High-Performance Timing Analysis Tool for VLSI Systems.

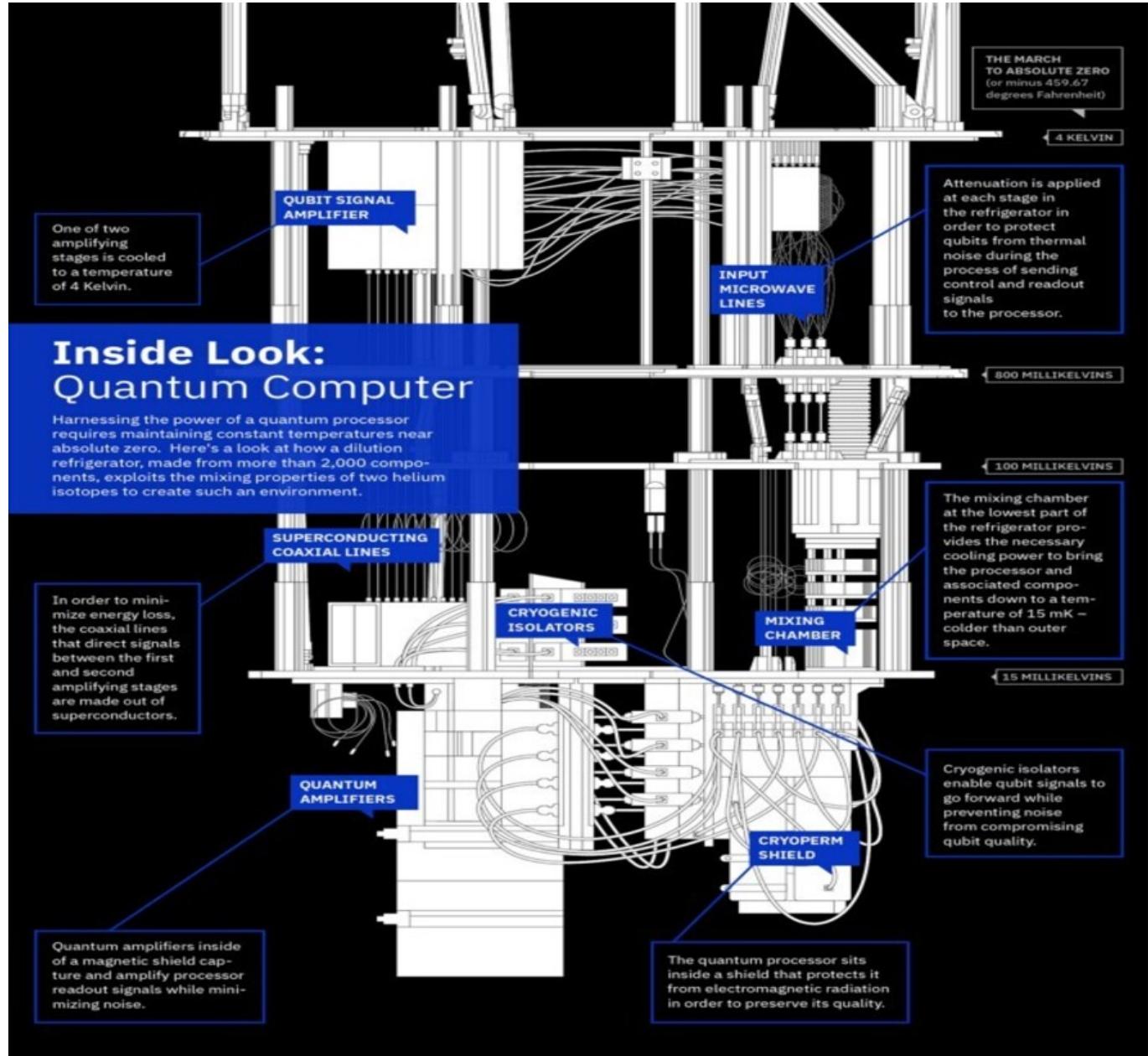
64. Quantum Computing

[Back to the Top](#)

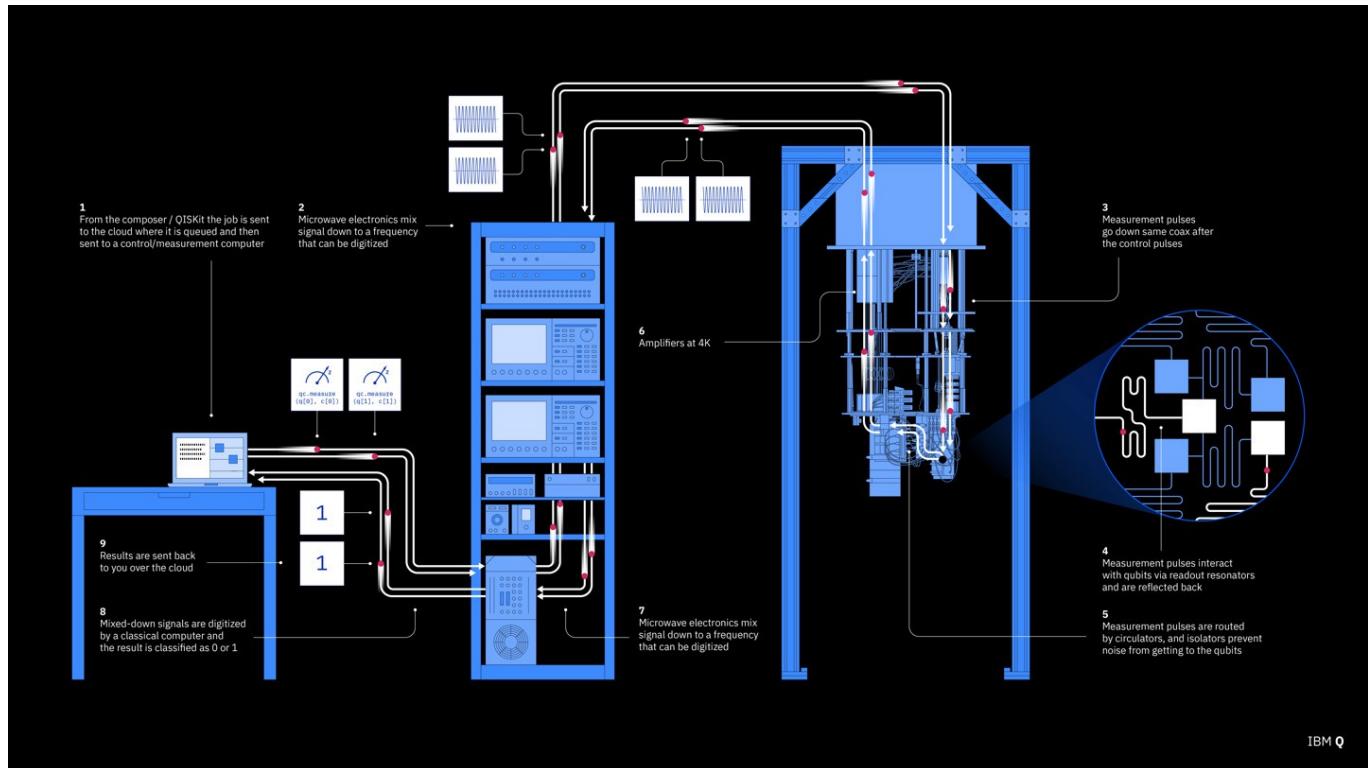


Quantum Computing Guide

A guide covering Quantum Computing including the applications, libraries and tools that will make you a better and more efficient developer in Quantum Computing.



A look inside of the IBM Q System One machine. Source: [IBM Research Quantum Experience, 2020](#).



IBM Q System One machine Overview. Source: [IBM Research Quantum Experience, 2020](#).

Resources

[Quantum computing](#) is the use of principles in quantum mechanics such as [superposition](#) and [entanglement](#) to process information. Quantum computing will play a big role in the innovation of artificial intelligence and machine learning.

[Qubits \(quantum bits\)](#) are the fundamental object of information in quantum computing.

[Introduction to Quantum computing](#)

[The Q# User Guide for Microsoft Quantum](#)

[Quantum computing research from Microsoft](#)

[Azure Quantum](#)

[Amazon Quantum Solutions Lab for Cloud Computing Services](#)

[Getting Started with Quantum Computing on Amazon Braket](#)

[Quantum Computing research from Google](#)

[NASA's Quantum Artificial Intelligence Laboratory \(QuAIL\)](#)

[TensorFlow Quantum: A Software Framework for Quantum Machine Learning](#)

[Hybrid quantum-classical Neural Networks with PyTorch and Qiskit](#)

[Supercomputing Solutions for AI and Research from NVIDIA](#)

[Quantum Computing research from Intel](#)

[Quantum Computing research from D-Wave Systems](#)

[Quantum Computing research from IonQ](#)

[IBM Quantum Experience](#)

[Quantum Computing Training and Courses on IBM Skills](#)

[Virtual Distillation for Quantum Error Mitigation](#)

[Power of data in quantum machine learning](#)

[More Efficient Quantum Computations of Chemistry through Tensor Hyper-Contraction](#)

[Demonstrating a Continuous Set of Two-qubit Gates for Near-term Quantum Algorithms](#)

[Decoding Quantum Errors Using Subspace Expansions](#)

[Quantum Computing Fundamentals online program from MIT](#)

[Online introductory lectures on quantum computing from CERN](#)

[QubitxQubit: quantum education program for K-12 students and beyond](#)

[Quantum Computing Courses on Coursera](#)

[Learn Quantum Computing with Online Courses from edX](#)

[Introduction to Quantum Computing on Udemy](#)

[Quantum Computing & Quantum Physics for Beginners on Udemy](#)

Tools

[Quantum development kit](#) is an open-source development kit from Microsoft to develop quantum applications and solve optimization problems. It includes the high-level quantum programming language Q#, a set of libraries, simulators, support for Q# in environments like Visual Studio Code and Jupyter Notebooks, and interoperability with Python or .NET languages.

[Quantum Katas](#) are a collection of self-paced tutorials and programming exercises from Microsoft to help you learn quantum computing and Q# programming.

[Visual Studio](#) is an integrated development environment (IDE) from Microsoft; which is a feature-rich application that can be used for many aspects of software development. Visual Studio makes it easy to edit, debug, build, and publish your app. By using Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, and Windows Store.

[Amazon Braket](#) is a fully managed quantum computing service that helps researchers and developers explore potential applications and evaluate current quantum computing technologies. It provides a development environment to design quantum algorithms, test them on simulated quantum computers, and run them on different types of quantum computing hardware.

[TensorFlow Quantum \(TFQ\)](#) is a quantum machine learning library for rapid prototyping of hybrid quantum-classical ML models.

[Mellanox Quantum™](#) is the world's smartest network switch silicon, designed to enable in-network computing through the Co-Design Scalable Hierarchical Aggregation and Reduction Protocol (SHARP) technology.

[Qiskit](#) is an open-source SDK for working with quantum computers at the level of circuits, algorithms, and application modules.

[Terra](#) is the foundation that the Qiskit SDK is built on. It allows the user to write quantum circuits easily, and takes care of the constraints of real hardware.

[Cirq](#) is a Python library for writing, manipulating, and optimizing Noisy Intermediate Scale Quantum (NISQ) circuits and running them against quantum computers and simulators.

[PyQuil](#) is a Python library for quantum programming using [Quil](#), the quantum instruction language developed at [Rigetti Computing](#).

[OpenFermion](#) is an open source library for compiling and analyzing quantum algorithms to simulate fermionic systems, including quantum chemistry.

[QuTiP](#) is open-source software for simulating the dynamics of closed and open quantum systems. The QuTiP library uses the excellent Numpy, Scipy, and Cython packages as the numerical backend, and graphical output is provided by Matplotlib.

[ProjectQ](#) is an open source software framework for quantum computing.

[Strawberry Fields](#) is a full-stack Python library for designing, simulating, and optimizing continuous variable (CV) quantum optical circuits.

[PennyLane](#) is a cross-platform Python library for [differentiable programming](#) of quantum computers. By training a quantum computer the same way as a neural network.

[PySyft](#) is a Python library for secure and private Deep Learning. PySyft decouples private data from model training, using [Federated Learning](#), [Differential Privacy](#), and Encrypted Computation (like [Multi-Party Computation \(MPC\)](#) and [Homomorphic Encryption \(HE\)](#)) within the main Deep Learning frameworks like PyTorch and [TensorFlow](#).

[BoTorch](#) is a library for Bayesian Optimization built on PyTorch.

[PyTorch Geometric \(PyG\)](#) is a geometric deep learning extension library for [PyTorch](#).

[Skorch](#) is a scikit-learn compatible neural network library that wraps PyTorch.

65. Data Structures & Algorithms

[Back to the Top](#)



Data Structures & Algorithms Guide

A guide covering Data Structures and Algorithms such as Arrays, Linked Lists, Trees, Hashtables, Stacks, Queues, Heaps, Graphs, Sorting & Searching, and Dynamic programming.

Resources

[VisualGo: Visualising data structures and algorithms through animation](#)

[Data Structure & Algorithms Visualizations](#)

[Learning Data Structures and Algorithms courses on Coursera](#)

[Learning Data Structures and Algorithms courses on Udemy](#)

[Learning Data Structures and Algorithms courses on edX](#)

[Intro to Data Structures and Algorithms courses on Udacity](#)

[Learning Algorithms courses on Khan Academy](#)

[Data Structures and Algorithms course from Harvard University](#)

[Data Structures and Algorithms course from Standford University\(CS 166\)](#)

[Learning Data Structures and Algorithms on CodeChef](#)

[Learning Algorithms and Data Structures courses on Pluralsight](#)

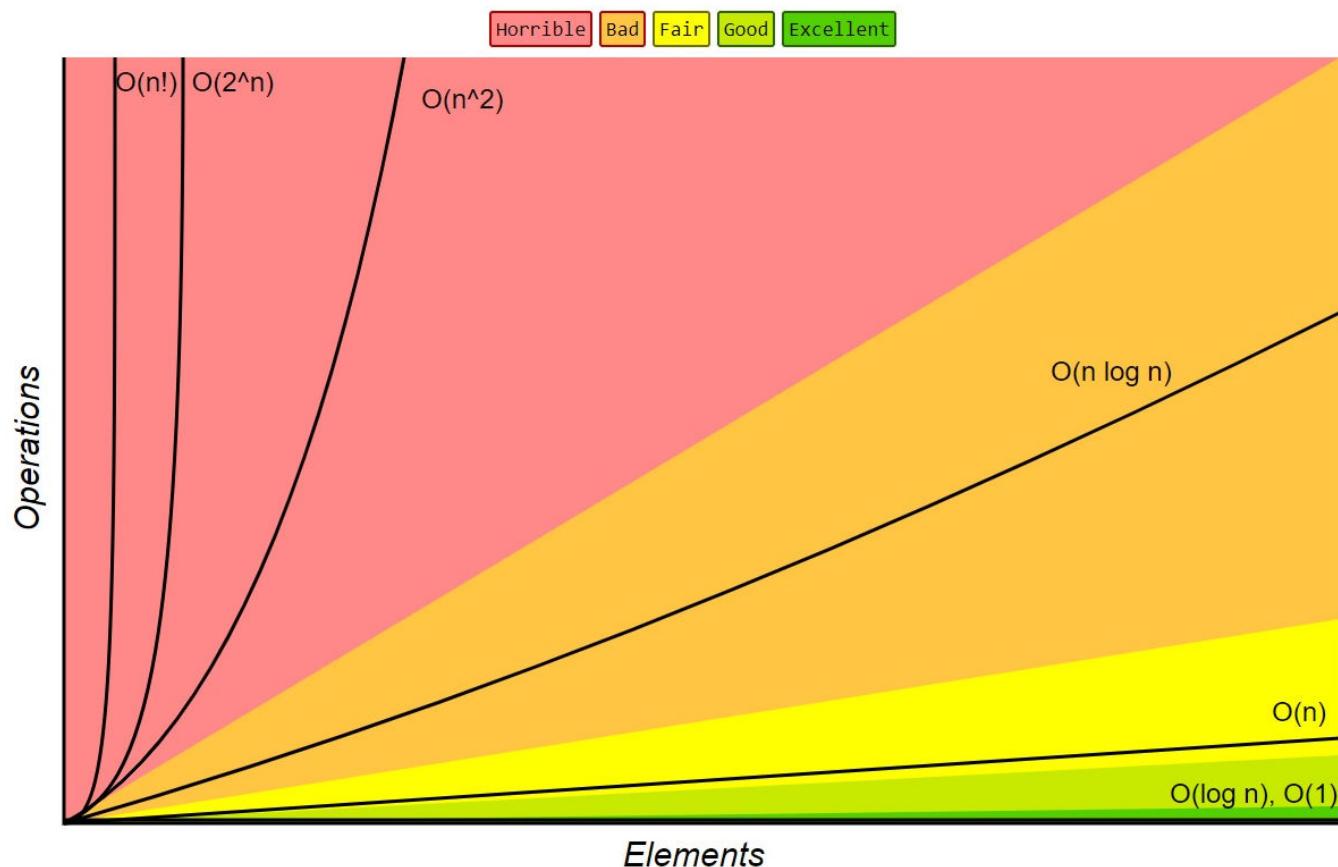
[Data structure and algorithms online course on Learnbay.io](#)

[Discrete Probability](#)

[Mathematical Proofs](#)

Algorithmic Complexity Analysis

- Big O Notation
- Time Complexity
- Space Complexity



Order of growth of algorithms specified in Big-O notation. Source: Big-O Cheat Sheet, 2016.

Data Structure	Time Complexity								Space Complexity	
	Average				Worst					
	Access	Search	Insertion	Deletion	Access	Search	Insertion	Deletion		
Array	O(1)	O(n)	O(n)	O(n)	O(1)	O(n)	O(n)	O(n)	O(n)	
Stack	O(n)	O(n)	O(1)	O(1)	O(n)	O(n)	O(1)	O(1)	O(n)	
Queue	O(n)	O(n)	O(1)	O(1)	O(n)	O(n)	O(1)	O(1)	O(n)	
Singly-Linked List	O(n)	O(n)	O(1)	O(1)	O(n)	O(n)	O(1)	O(1)	O(n)	
Doubly-Linked List	O(n)	O(n)	O(1)	O(1)	O(n)	O(n)	O(1)	O(1)	O(n)	
Skip List	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)	O(n)	O(n)	O(n)	O(n log(n))	
Hash Table	N/A	O(1)	O(1)	O(1)	N/A	O(n)	O(n)	O(n)	O(n)	
Binary Search Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)	O(n)	O(n)	O(n)	O(n)	
Cartesian Tree	N/A	O(log(n))	O(log(n))	O(log(n))	N/A	O(n)	O(n)	O(n)	O(n)	
B-Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)	
Red-Black Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)	
Splay Tree	N/A	O(log(n))	O(log(n))	O(log(n))	N/A	O(log(n))	O(log(n))	O(log(n))	O(n)	
AVL Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)	
KD Tree	O(log(n))	O(log(n))	O(log(n))	O(log(n))	O(n)	O(n)	O(n)	O(n)	O(n)	

Complexity of operations on Data Structures. Source: Big-O Cheat Sheet, 2016.

Algorithm	Time Complexity			Space Complexity
	Best	Average	Worst	Worst
<u>Quicksort</u>	$\Omega(n \log(n))$	$\Theta(n \log(n))$	$O(n^2)$	$O(\log(n))$
<u>Mergesort</u>	$\Omega(n \log(n))$	$\Theta(n \log(n))$	$O(n \log(n))$	$O(n)$
<u>Timsort</u>	$\Omega(n)$	$\Theta(n \log(n))$	$O(n \log(n))$	$O(n)$
<u>Heapsort</u>	$\Omega(n \log(n))$	$\Theta(n \log(n))$	$O(n \log(n))$	$O(1)$
<u>Bubble Sort</u>	$\Omega(n)$	$\Theta(n^2)$	$O(n^2)$	$O(1)$
<u>Insertion Sort</u>	$\Omega(n)$	$\Theta(n^2)$	$O(n^2)$	$O(1)$
<u>Selection Sort</u>	$\Omega(n^2)$	$\Theta(n^2)$	$O(n^2)$	$O(1)$
<u>Tree Sort</u>	$\Omega(n \log(n))$	$\Theta(n \log(n))$	$O(n^2)$	$O(n)$
<u>Shell Sort</u>	$\Omega(n \log(n))$	$\Theta(n(\log(n))^2)$	$O(n(\log(n))^2)$	$O(1)$
<u>Bucket Sort</u>	$\Omega(n+k)$	$\Theta(n+k)$	$O(n^2)$	$O(n)$
<u>Radix Sort</u>	$\Omega(nk)$	$\Theta(nk)$	$O(nk)$	$O(n+k)$
<u>Counting Sort</u>	$\Omega(n+k)$	$\Theta(n+k)$	$O(n+k)$	$O(k)$
<u>Cubesort</u>	$\Omega(n)$	$\Theta(n \log(n))$	$O(n \log(n))$	$O(n)$

Complexity of Sorting algorithms. Source: Big-O Cheat Sheet, 2016.

Data Structures

Arrays are a series of elements(numbers, booleans, or strings) of the same type placed in physical memory locations that can be individually referenced by adding an index to a unique identifier.

- **Vector** is a dynamic array, where the size can be increased when an element is inserted or deleted, with the storage being handled automatically by the container.

Linked Lists is a linear collection of data elements whose order is not given by their physical placement in memory. Instead, elements in a linked list are linked using pointers.

- **Singly Linked List** is a type of linked data structure where each node points to the next node in the sequence. It does not have any pointer that points to the previous node.
- **Doubly-Linked List** is a type of linked data structure in which each node apart from storing its data has two links. A node consists of three parts: node data, a pointer to the next node in sequence (next pointer), pointer to the previous node (previous pointer).

Trees is a nonlinear data structure unlike arrays, linked lists, stacks and queues which are linear data structures. A tree can be empty with no nodes or in other instances a tree's structure consists of one node called the root and 0 or 1 or more subtrees.

- Basic Tree
- Binary Tree

- Binary Search Tree
- AVL Tree
- Red-Black Tree
- N-ary Tree

[Hash Tables](#) is a data structure, much like an array, except you store each value (object) using a key. It is a basic key/value store for mapping known as a hash function.

[Stacks](#) is a linear data structure that stores data in an order known as the Last In First Out (LIFO) order.

- LIFO (Last In First Out)
- FILO (First In Last Out)

[Queues](#) is a linear data structure that stores data in an order known as the First In First Out (FIFO) order.

[Heaps](#) is a specialized tree-based data structure which is essentially an almost complete tree that satisfies the heap property.

- Max Heap
- Min-Heap
- Binary Heap

[Graphs](#) is a data structure that represent relations between pairs of objects. It consists of nodes (known as vertices) that are connected through links (known as edges).

Algorithms

Graph Algorithms

- Breadth First Search (BFS)
- Depth First Search (DFS)
- Shortest Path from source to all vertices **Dijkstra**
- Shortest Path from every vertex to every other vertex **Floyd Warshall**
- Minimum Spanning tree **Prim**
- Minimum Spanning tree **Kruskal**
- Topological Sort
- Articulation Points (Cut Vertices) in a Graph
- Bridges in a graph

Searching And Sorting

- Binary Search
- Quick Sort
- Merge Sort
- Order Statistics
- KMP algorithm
- Rabin karp
- Z's algorithm
- String Matching/String Parsing
- Counting Sort

Dynamic Programming

- Longest Common Subsequence
- Longest Increasing Subsequence
- Edit Distance
- Minimum Partition
- Ways to Cover a Distance
- Longest Path In Matrix
- Subset Sum Problem
- Optimal Strategy for a Game
- 0-1 Knapsack Problem
- Assembly Line Scheduling

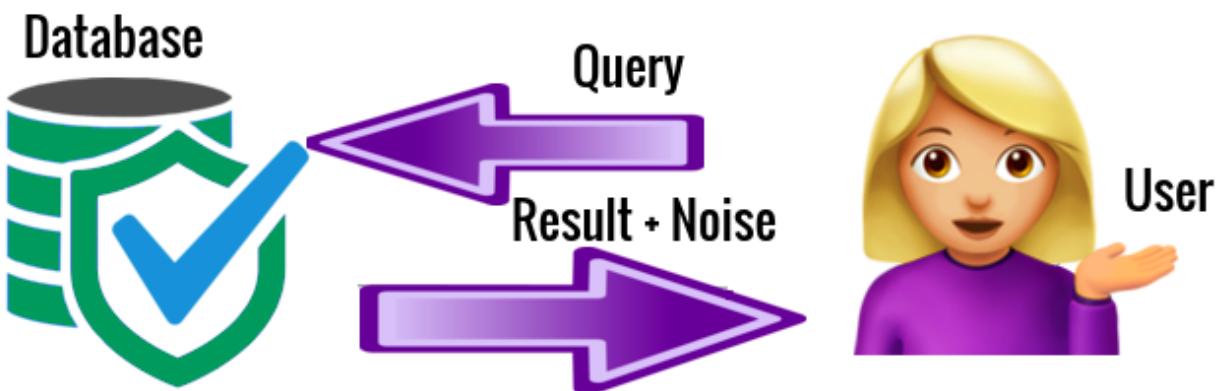
66. Differential Privacy

[Back to the Top](#)



Differential Privacy Guide

A guide covering Differential Privacy including the applications, libraries and tools that will make you a better and more efficient developer with protecting users data and their privacy.



Above is a simple diagram of how Differential Privacy-Preserving Data Sharing and Data Mining protects a User's Data

Resources

[Differential Privacy](#) is a system that simultaneously enables researchers and analysts to extract useful insights from datasets containing personal information and offers stronger privacy protections. This is achieved by introducing "statistical noise".

[Statistical Noise](#) is a process that small aletrations to masked datasets. The statistical noise hides identifiable characteristics of individuals, ensuring that the privacy of personal information is protected, but it's small enough to not materially impact the accuracy of the answers extracted by analysts and researchers.

[Laplacian Noise](#) is a mechanism that adds Laplacian-distributed noise to a function.

[Differential Privacy Blog Series by the National Institute of Standards and Technology\(NIST\)](#)

[Apple's Differential Privacy Overview](#)

[Learning with Privacy at Scale with Apple Machine Learning](#)

[Microsoft Research Differential Privacy Overview](#)

[Responsible Machine Learning with Microsoft Azure](#)

[Responsible AI Resources with Microsoft AI](#)

[Preserve data privacy by using differential privacy and the SmartNoise package](#)

[Open Differential Privacy\(OpenDP\) Initiative by Microsoft and Harvard](#)

[Google's Differential Privacy Library](#)

[Computing Private Statistics with Privacy on Beam from Google Codelabs](#)

[Introducing TensorFlow Privacy: Learning with Differential Privacy for Training Data](#)

[TensorFlow Federated: Machine Learning on Decentralized Data](#)

[Federated Analytics: Collaborative Data Science without Data Collection](#)

[Differentially-Private Stochastic Gradient Descent\(DP-SGD\)](#)

[Learning Differential Privacy from Harvard University Privacy Tools Project](#)

[Harvard University Privacy Tools Project Courses & Educational Materials](#)

[The Weaknesses of Differential Privacy course on Coursera](#)

[The Differential Privacy of Bayesian Inference](#)

[Simultaneous private learning of multiple concepts](#)

[The Complexity of Computing the Optimal Composition of Differential Privacy](#)

[Order revealing encryption and the hardness of private learning](#)

[SAP HANA data anonymization using SAP Software Solutions](#)

[SAP HANA Security using their In-Memory Database](#)

[DEFCON Differential Privacy Training Launch](#)

[Secure and Private AI course on Udacity](#)

[Differential Privacy - Security and Privacy for Big Data - Part 1 course on Coursera](#)

[Differential Privacy - Security and Privacy for Big Data - Part 2 course on Coursera](#)

[Certified Ethical Emerging Technologist Professional Certificate course on Coursera](#)

Tools

[PySyft](#) is a Python library for secure and private Deep Learning. PySyft decouples private data from model training, using [Federated Learning](#), [Differential Privacy](#), and Encrypted Computation (like [Multi-Party Computation \(MPC\)](#) and [Homomorphic Encryption \(HE\)](#)) within the main Deep Learning frameworks like [PyTorch](#) and [TensorFlow](#).

[TensorFlow Privacy](#) is a Python library that includes implementations of TensorFlow optimizers for training machine learning models with differential privacy. The library comes with tutorials and analysis tools for computing the privacy guarantees provided.

[TensorFlow Federated \(TFF\)](#) is an open-source framework for machine learning and other computations on decentralized data. TFF has been developed to facilitate open research and experimentation with [Federated Learning \(FL\)](#), an approach to machine learning where a shared global model is trained across many participating clients that keep their training data locally.

[Privacy on Beam](#) is an end-to-end differential privacy solution built on [Apache Beam](#). It is intended to be usable by all developers, regardless of their differential privacy expertise.

[PyDP](#) is a Python wrapper for Google's Differential Privacy project.

[PennyLane](#) is a cross-platform Python library for [differentiable programming](#) of quantum computers. By training a quantum computer the same way as a neural network.

[BoTorch](#) is a library for Bayesian Optimization built on PyTorch.

[PyTorch Geometric \(PyG\)](#) is a geometric deep learning extension library for [PyTorch](#).

[Skorch](#) is a scikit-learn compatible neural network library that wraps PyTorch.

[Diffprivlib](#) is the IBM Differential Privacy Library for experimenting with, investigating and developing applications in, differential privacy.

[Opacus](#) is a library that enables training PyTorch models with differential privacy. It supports training with minimal code changes required on the client, has little impact on training performance and allows the client to online track the privacy budget expended at any given moment.

[Smart Noise](#) is a toolkit that uses state-of-the-art differential privacy (DP) techniques to inject noise into data, to prevent disclosure of sensitive information and manage exposure risk.

67. Raspberry Pi

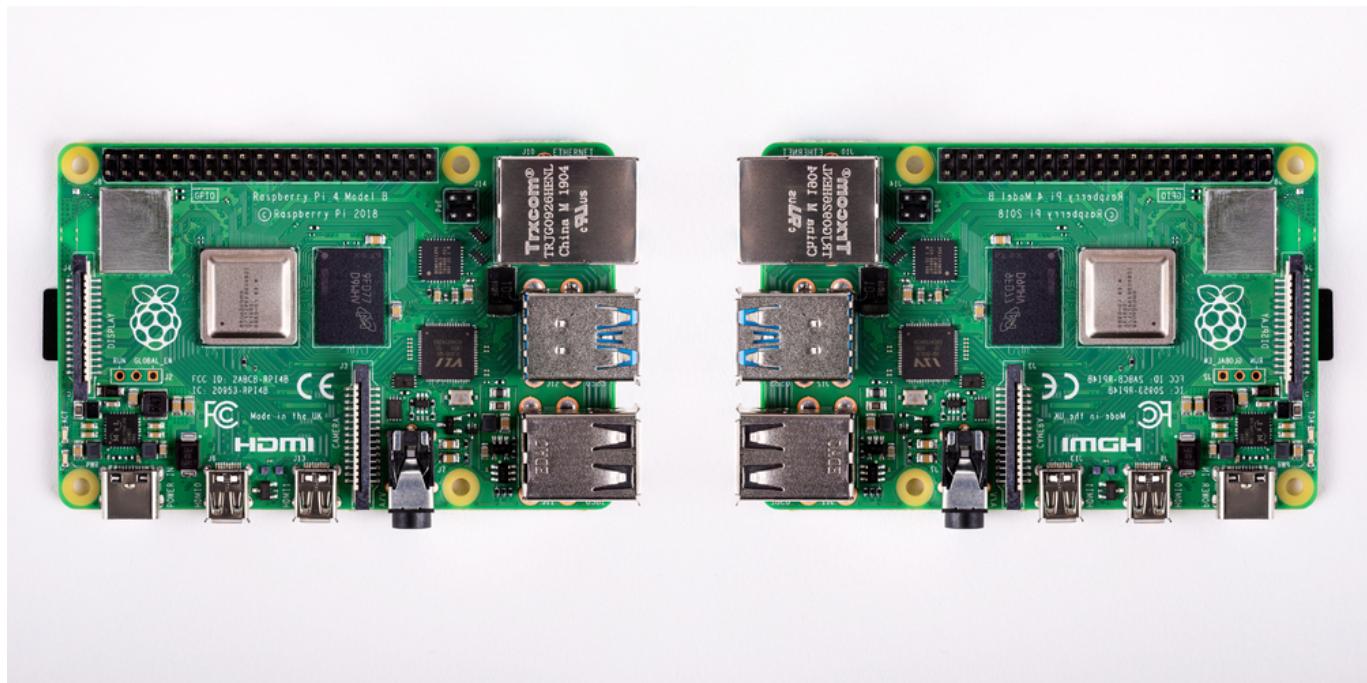
[Back to the Top](#)



Raspberry Pi Guide

A guide covering Raspberry Pi devices such as the Raspberry Pi 4 Model B and Raspberry Pi 400. Along with a wide variety of operating systems that you can install on your Raspberry Pi device. Also, learn about cool projects that you can build with your Raspberry Pi device.

Raspberry Pi 4 Model B



[Check out the Raspberry Pi 4](#)

Raspberry Pi 4 Model B Hardware Specifications

- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- 2GB, 4GB or 8GB LPDDR4-3200 SDRAM (depending on model)
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless
- Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 USB 3.0 ports; 2 USB 2.0 ports.
- Raspberry Pi standard 40 pin GPIO header (fully backwards compatible with previous Pi boards)
- 2 × micro-HDMI ports (up to 4kp60 supported)
- OpenGL ES 3.0 graphics

Raspberry Pi 400 Personal Computer Kit

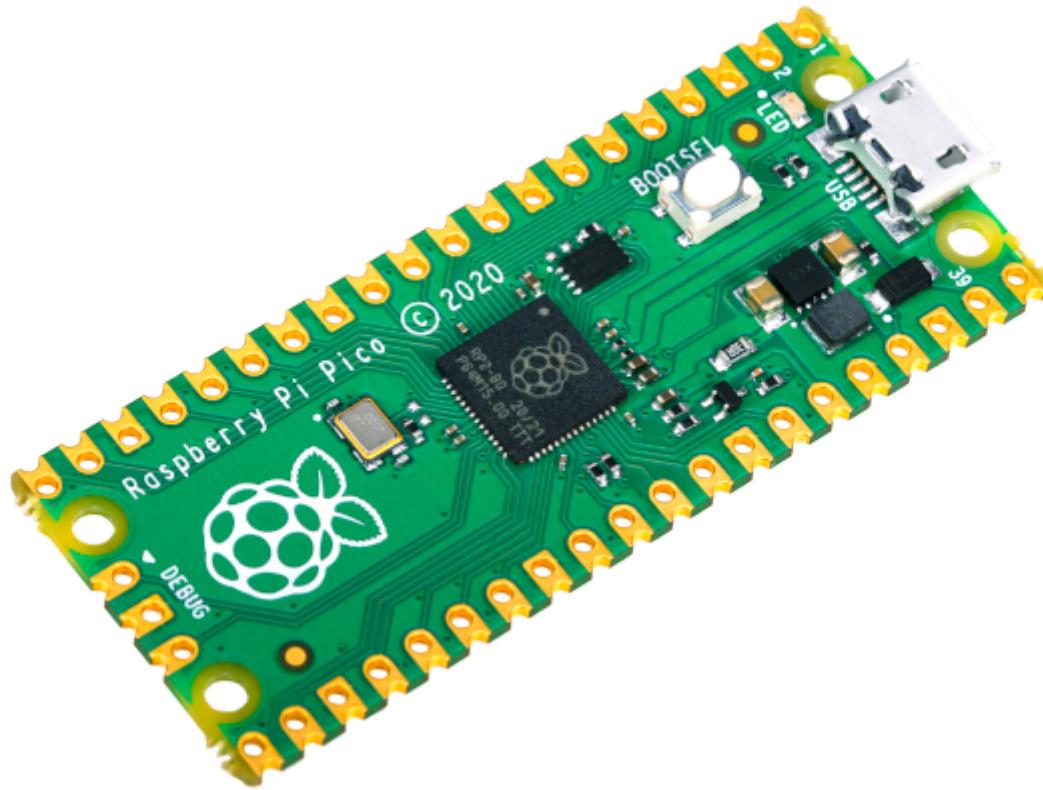


[Check out the Raspberry Pi 400 Personal Computer Kit](#)

Raspberry Pi 400 Hardware Specifications

- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.8GHz
- 4GB LPDDR4-3200 SDRAM
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless
- Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 USB 3.0 ports; 2 USB 2.0 ports.
- Raspberry Pi standard 40 pin GPIO header
- 2 × micro-HDMI ports (up to 4kp60 supported)
- OpenGL ES 3.0 graphics

Raspberry Pi Pico microcontroller



[Check out the Raspberry Pi Pico](#)

Raspberry Pi Pico Hardware Specifications

- RP2040 microcontroller chip designed by Raspberry Pi in the UK
- Dual-core Arm Cortex-M0+ processor, flexible clock running up to 133 MHz
- 264KB on-chip SRAM
- 2MB on-board QSPI Flash
- 26 multifunction GPIO pins, including 3 analogue inputs
- 2 × UART, 2 × SPI controllers, 2 × I2C controllers, 16 × PWM channels
- 1 × USB 1.1 controller and PHY, with host and device support
- 8 × Programmable I/O (PIO) state machines for custom peripheral support
- Castellated module allows soldering direct to carrier boards
- Drag-and-drop programming using mass storage over USB
- Low-power sleep and dormant modes
- Accurate on-chip clock
- Temperature sensor
- Accelerated integer and floating-point libraries on-chip

Raspberry Pi OS. The default Operating System for every Raspberry Pi device

[Check out Raspberry Pi OS](#)



Resources

[Raspberry Pi](#) is an ARM powered single board computer(SBC) that is the size of a credit card and costs around \$35.

[Raspberry Pi Foundation](#) is a UK-based charity that works to put the power of computing and digital making into the hands of people all over the world.

[Getting Started with Raspberry Pi Projects](#)

[Online learning for the Raspberry Pi](#)

[Raspberry Pi Training Program](#)

[Raspberry Pi Online Courses on Udemy](#)

[Raspberry Pi Online Courses on Coursera](#)

[The Raspberry Pi Platform and Python Programming course on Coursera](#)

[Learning Raspberry Pi with Online Courses on edX](#)

[Raspberry Pi Online Training Courses on LinkedIn Learning](#)

[Getting Started with Raspberry Pi course on FutureLearn](#)

[Home Assistant on Raspberry Pi](#)

[PiSwitch: Build your own Nintendo Switch-style console](#)

Raspberry Pi Operating Systems

[Raspberry Pi OS](#)

[Hass.io\(Home Assistant OS\)](#)

[OmniROM\(Android 11\) based on ASOP](#)

[Manjaro Linux ARM](#)

[Arch Linux ARM](#)

[Ubuntu MATE for Raspberry Pi](#)

[Ubuntu Desktop for Raspberry Pi](#)

[Ubuntu Core on a Raspberry Pi](#)

[Ubuntu Server for ARM](#)

[Fedora ARM](#)

[Kali Linux for the Raspberry Pi](#)

[Twister OS](#)

[TitusPi](#)

[RetroArch](#)

[RetroPie](#)

[LibreELEC](#)

[OSMC](#)

[RISC OS](#)

[DietPi](#)

[Windows 10 IoT Core](#)

Tools

[Raspberry Pi Imager](#) is the quick and easy way to install Raspberry Pi OS and other operating systems to a microSD card, ready to use with your Raspberry Pi.

[Etcher](#) is an open source, cross-platform software that makes it easy to flash operating system images to a microSD card or USB device.

[Home Assistant](#) is an open source home automation that puts local control and privacy first. Home Assistant is powered by a worldwide community of tinkerers and DIY enthusiasts that runs great on Raspberry Pi.

[Gladys Assistant](#) is a privacy-first, open-source home assistant and runs great on Raspberry Pi.

[Kodi for Raspberry Pi](#) is a free and open source media player application developed by the XBMC/Kodi Foundation.

[Pi-hole](#) is a DNS sinkhole that protects your devices from unwanted content, without installing any client-side software, intended for use on a private network. It is designed for use on embedded devices with network capability, such as the Raspberry Pi, but it can be used on other machines running Linux and cloud implementations.

[PiKVM](#) is a very simple and fully functional Raspberry Pi-based KVM over IP.

[PiShrink](#) is a bash script that automatically shrink a pi image that will then resize to the max size of the SD card on boot.

[RPiPlay](#) is an open-source implementation of an AirPlay mirroring server for the Raspberry Pi that supports iOS 9 and later.

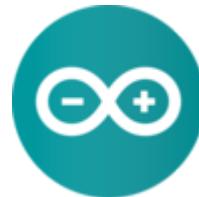
[Gpiozero](#) is a simple interface to GPIO(General-Purpose Input/Output) devices with the Raspberry Pi.

[Balena Sound](#) is a single or multi-room streamer for an existing audio device using a Raspberry Pi! It supports Bluetooth, Airplay and Spotify Connect.

[OpenBalena](#) is a platform to deploy and manage connected devices.

68. Arduino

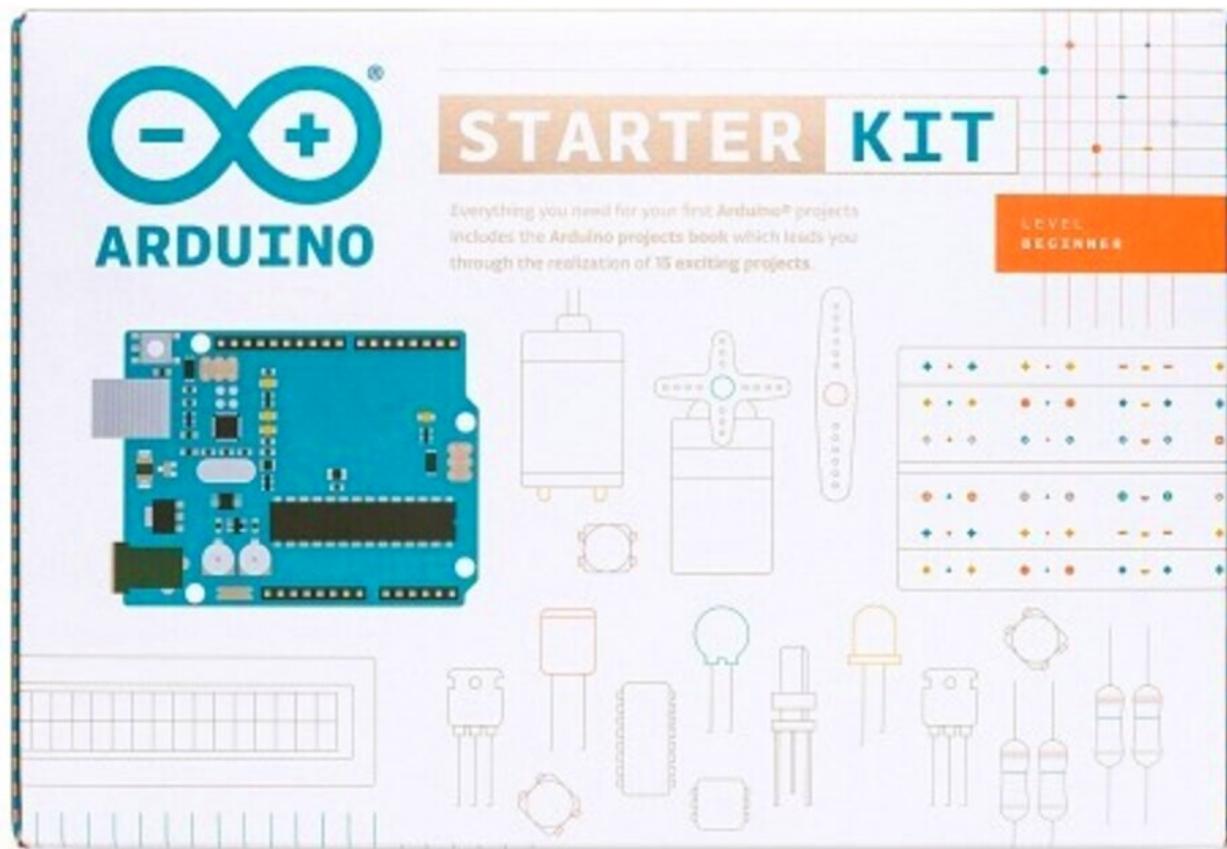
[Back to the Top](#)



Arduino Guide

A guide covering Arduino devices such as the Arduino Uno and Arduino Nano. Along with the tools, applications and libraries that will make you a better and more efficient developer with Arduino devices. Also, learn about cool projects that you can build with your Arduino device.

[Checkout the Arduino Starter Kit](#)



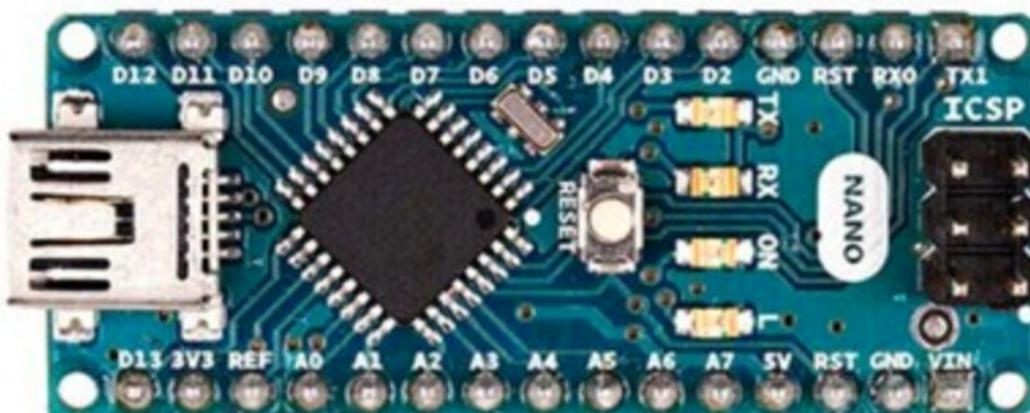
[Checkout the Arduino Uno](#)



Arduino Uno Hardware Specs

- Microcontroller ATmega328P
- Operating Voltage 5V
- Input Voltage (recommended) 7-12V
- Input Voltage (limit) 6-20V
- Digital I/O Pins 14 (of which 6 provide PWM output)
- Flash Memory 32 KB (ATmega328P) of which 0.5 KB used by bootloader
- SRAM 2 KB (ATmega328P)
- EEPROM 1 KB (ATmega328P)
- Clock Speed 16 MHz

[Checkout the Arduino Nano](#)



Arduino Nano Hardware Specs

- Microcontroller ATmega328
- Architecture AVR
- Operating Voltage 5 V
- Flash Memory 32 KB of which 2 KB used by bootloader
- SRAM 2 KB
- Clock Speed 16 MHz
- Input Voltage 7-12 V
- Digital I/O Pins 22 (6 of which are PWM)

Resources

[Arduino](#) is an open source electronic prototyping platform enabling users to create interactive electronic objects.

[Arduino Reference Manual](#)

[Getting Started with Arduino](#)

[Arduino Education](#)

[Arduino Store](#)

[Arduino Forum](#)

[Arduino Courses on Udemy](#)

[Arduino Courses on Coursera](#)

[Online Arduino Classes on Skillshare](#)

[Introduction to Arduino course on Pluralsight](#)

[Arduino Training Courses on NobleProg](#)

[Arduino with Python course from Real Python](#)

[200+ Arduino Projects on Instructables](#)

Tools

[Arduino CLI](#) is an all-in-one solution that provides builder, Boards/Library Manager, uploader, discovery and many other tools needed to use any Arduino compatible board and platforms.

[Arduino IDE](#) is an open source integrated development environment(IDE) for the Arduino computing platforms based on easy-to-use hardware and software.

[Arduino Pro IDE](#) is a professional version of the Arduino IDE that brings more features for advanced users and developers. The project is currently in the experimently version.

[VSCode Arduino](#) is a VSCode extension for Arduino. The Arduino extension makes it easy to develop, build, deploy and debug your Arduino sketches in Visual Studio Code, with a rich set of functionalities.

[Gobot](#) is a framework using the [Golang](#) programming language for robotics, physical computing, and the Internet of Things(IoT).

[Tasmota](#) is an alternative firmware for [ESP8266](#) based devices with easy configuration using webUI, OTA updates, automation using timers or rules, expandability and entirely local control over MQTT, HTTP, Serial or KNX.

[Blynk](#) is a C++ library for embedded hardware. It works with over 400 hardware models such as Arduino, ESP8266, Raspberry Pi, Intel Edison/Galileo, LinkIt ONE, Particle Core/Photon, ARM mbed, etc.

[Arduino Thread](#) is a library tha makes it simple to run Threads on Arduino.

[PlatformIO](#) is a professional collaborative platform for embedded development with no vendor lock-in. It provides support for multiplatforms and frameworks such as IoT, Arduino, CMSIS, ESP-IDF, FreeRTOS, libOpenCM3, mbed OS, Pulp OS, SPL, STM32Cube, Zephyr RTOS, ARM, AVR, Espressif (ESP8266/ESP32), FPGA, MCS-51 (8051), MSP430, Nordic (nRF51/nRF52), NXP i.MX RT, PIC32, RISC-V.

[PlatformIO for VSCode](#) is a plugin that provides support for the PlatformIO IDE on VSCode.

[Tock](#) is an embedded operating system designed for running multiple concurrent, mutually distrustful applications on Cortex-M and RISC-V based embedded platforms. Tock's design centers around protection, both from potentially malicious applications and from device drivers.

[TinyGo](#) is a Go compiler(based on LLVM) intended for use in small places such as microcontrollers, WebAssembly (Wasm), and command-line tools.

[LLVM](#) is a library that has collection of modular/reusable compiler and toolchain components (assemblers, compilers, debuggers, etc.). With these components LLVM can be used as a compiler framework, providing a front-end(parser and lexer) and a back-end (code that converts LLVM's representation to actual machine code).

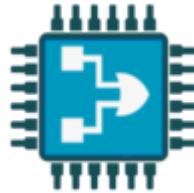
[Chipyard](#) is an open source framework for agile development of Chisel-based systems-on-chip. It will allow you to leverage the Chisel HDL, Rocket Chip SoC generator, and other [Berkeley](#) projects to produce a RISC-V SoC with everything from MMIO-mapped peripherals to custom accelerators.

[The Eclipse Embedded CDT](#) is a collection of plug-ins for Arm & RISC-V C/C++ developers. [Unicorn](#) is a lightweight, multi-platform, multi-architecture CPU emulator framework(ARM, AArch64, M68K, Mips, Sparc, X86) based on [QEMU](#).

[Keystone](#) is a lightweight multi-platform, multi-architecture(Arm, Arm64, Hexagon, Mips, PowerPC, Sparc, SystemZ & X86) assembler framework.

69. FPGA

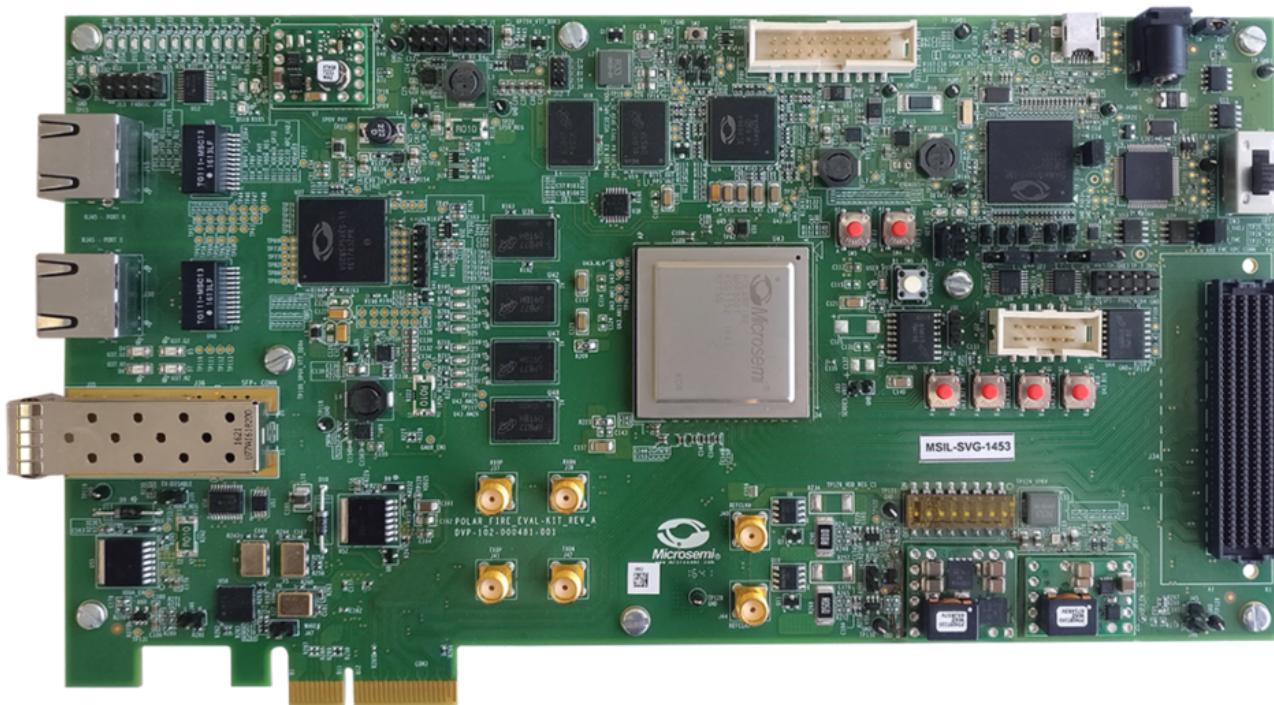
[Back to the Top](#)



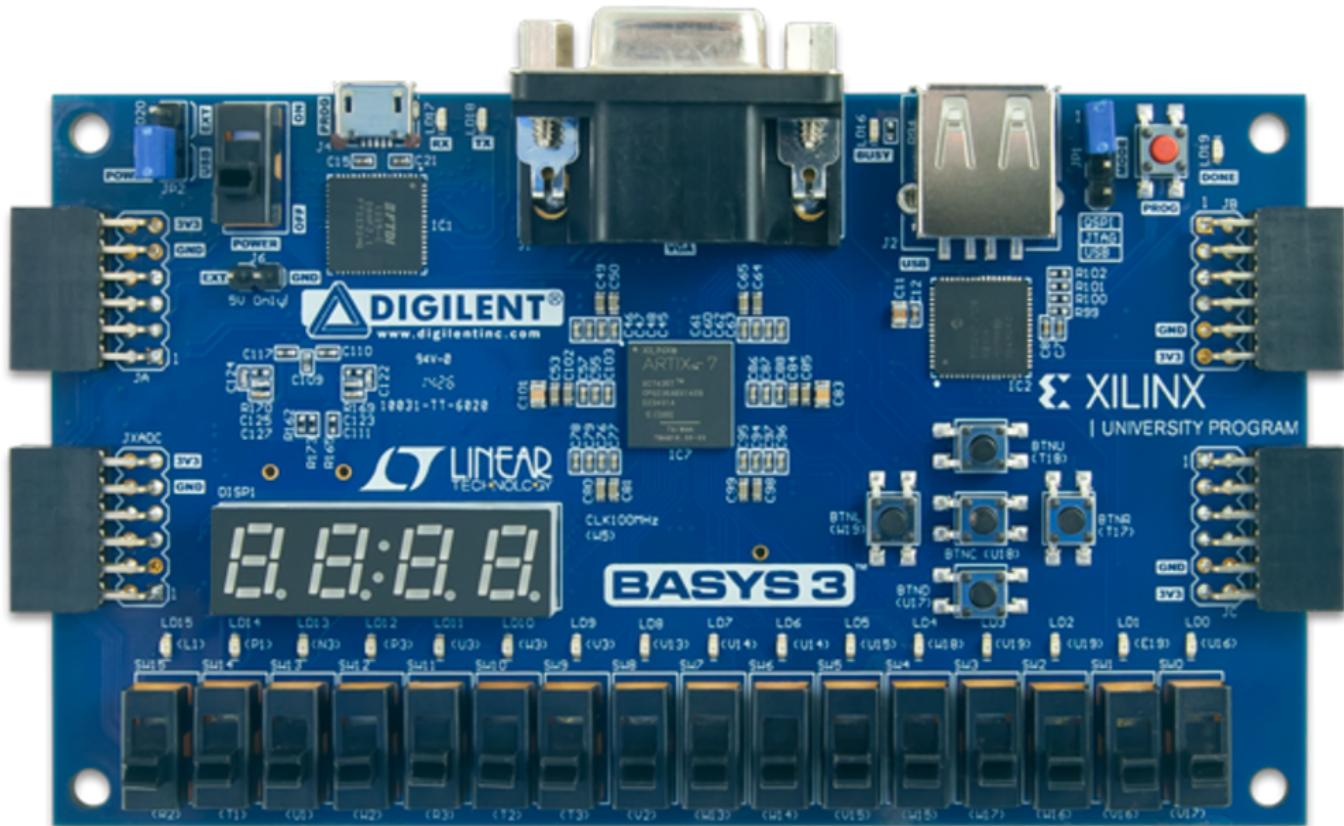
FPGA Guide

A guide covering **FPGA(Field Programmable Gate Arrays)** devices such as the **PolarFire®**, **Artix 7**, **Spartan 6** and **Zynq-7000**. Along with the tools, applications and libraries that will make you a better and more efficient developer with FPGA devices. Also, learn about cool projects that you can build with your FPGA device.

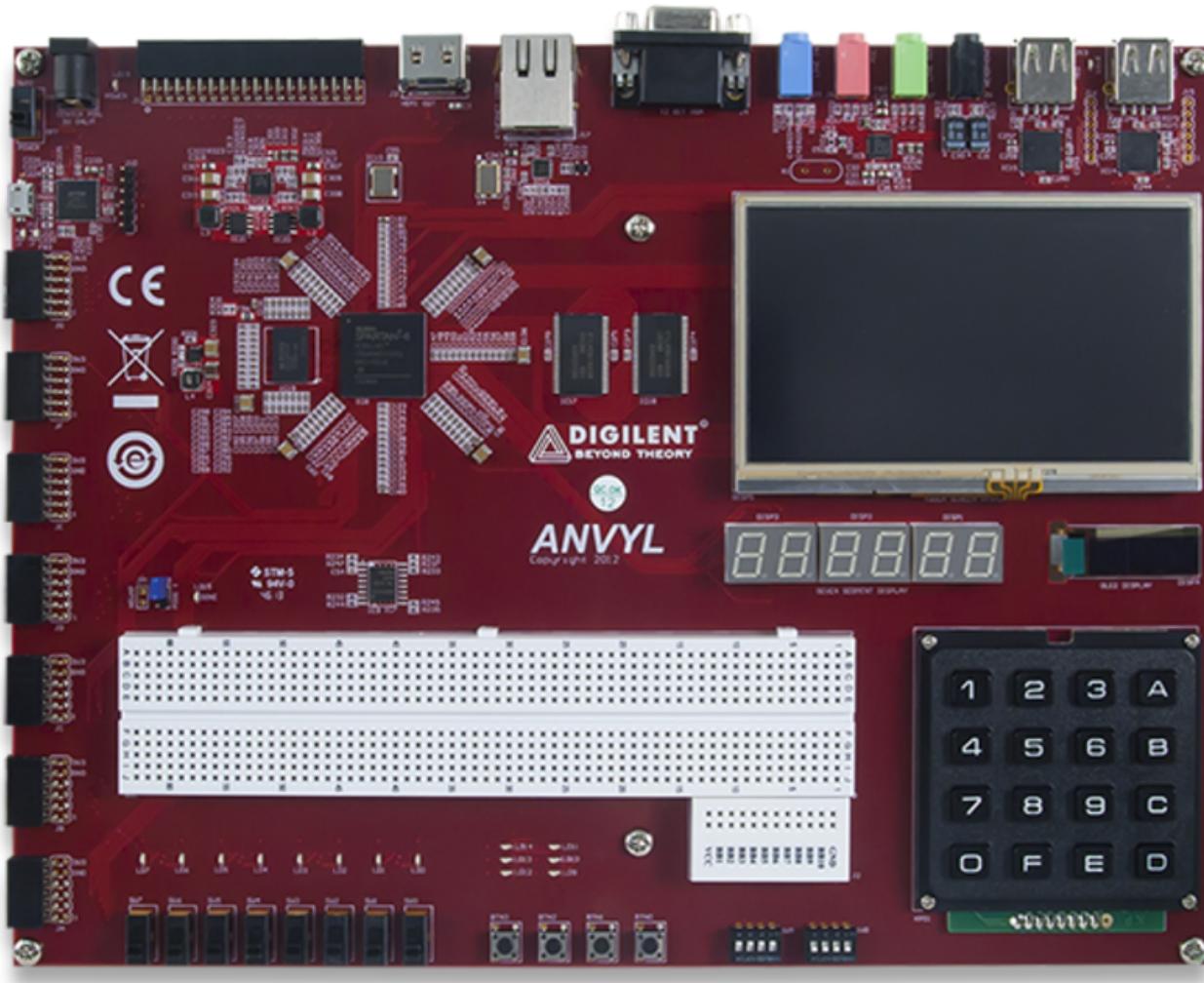
[Checkout the PolarFire® FPGA Development Kits](#)



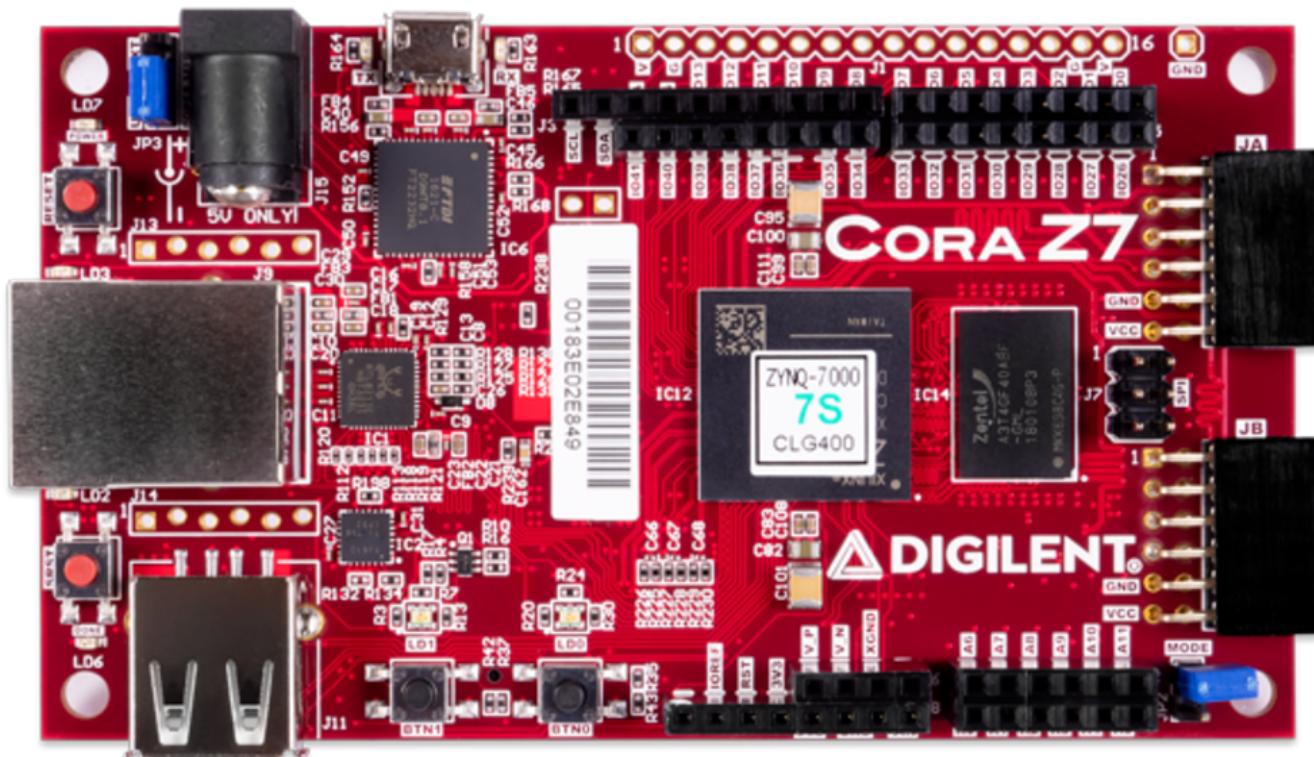
[Checkout the Artix 7 FPGA Development board](#)



[Checkout the Spartan 6 FPGA Development board](#)



[Checkout the Zynq-7000 for ARM/FPGA SoC Development board](#)



Resources

[FPGA\(Field Programmable Gate Arrays\)](#) are semiconductor devices that are based around a matrix of configurable logic blocks (CLBs) connected via programmable interconnects. FPGAs can be reprogrammed to desired application or functionality requirements after manufacturing.

[TinyFPGA](#) is a new series of boards that are low-cost, open source FPGA boards in a tiny form factor.

[SiFive FPGA shells](#)

[FPGA & SoC Design Tools from Microsemi](#)

[QuickLogic Embedded FPGA \(eFPGA\) Intellectual Property \(IP\) and Software](#)

[FPGA for Beginners with Development Boards from Digilent®](#)

[Hundreds of FPGA Projects on Instructables](#)

[FPGA Fundamentals from NI\(National Instruments\)](#)

[Getting Started With LabVIEW FPGA from NI\(National Instruments\)](#)

[Programming and FPGA Basics - INTEL® FPGAS](#)

[Intel FPGA Training Program](#)

[FPGA Courses on Coursera](#)

[FPGA Courses on Udemy](#)

[FPGA Online Training Courses on LinkedIn Learning](#)

[UMass Lowell's Graduate Certificate in Field Programmable Gate Arrays\(FPGA\)](#)

[FPGA Design Fundamentals Course \(UC San Diego Extension\)](#)

[FPGA II Course \(UC San Diego Extension\)](#)

[FPGAs & SoCs Training from Microsemi](#)

[DSP fundamentals for FPGAs course from MATLAB and Simulink Training](#)

[Verilog Courses on Coursera](#)

Tools

[LabVIEW FPGA](#) is a software add-on for LabVIEW that you can use to more efficiently and effectively design FPGA-based systems through a highly integrated development environment, IP libraries, a high-fidelity simulator, and debugging features.

[Apio](#) is a multiplatform toolbox, with static pre-built packages, project configuration tools and easy command interface to verify, synthesize, simulate and upload your verilog designs.

[IceStorm](#) is a project that aims at documenting the bitstream format of Lattice iCE40 FPGAs and providing simple tools for analyzing and creating bitstream files.

[Icestudio](#) is a visual editor for open FPGA boards. Built on top of the Icestorm project using Apio.

[FuseSoC](#) is an award-winning package manager and a set of build tools for HDL (Hardware Description Language) code and FPGA/ASIC development.

[OpenWiFi](#) is an open-source IEEE802.11/Wi-Fi baseband chip/FPGA design.

[PipeCNN](#) is an OpenCL-based FPGA Accelerator for Large-Scale Convolutional Neural Networks (CNNs). Currently, there is a growing trend among developers in the FPGA community to utilize High Level Synthesis (HLS) tools to design and implement customized circuits on FPGAs.

[Verilator](#) is an open-source SystemVerilog simulator and lint system.

[Verilog to Routing\(VTR\)](#) is a collaborative project to provide a open-source framework for conducting FPGA architecture and CAD Research & Development. The VTR design flow takes as input a Verilog description of a digital circuit, and a description of the target FPGA architecture.

[PlatformIO](#) is a professional collaborative platform for embedded development with no vendor lock-in. It provides support for multiplatforms and frameworks such as IoT, Arduino, CMSIS, ESP-IDF, FreeRTOS, libOpenCM3, mbed OS, Pulp OS, SPL, STM32Cube, Zephyr RTOS, ARM, AVR, Espressif (ESP8266/ESP32), FPGA, MCS-51 (8051), MSP430, Nordic (nRF51/nRF52), NXP i.MX RT, PIC32, RISC-V.

[PlatformIO for VSCode](#) is a plugin that provides support for the PlatformIO IDE on VSCode.

[Tock](#) is an embedded operating system designed for running multiple concurrent, mutually distrustful applications on Cortex-M and RISC-V based embedded platforms. Tock's design centers around protection, both from potentially malicious applications and from device drivers.

[OpenTimer](#) is a High-Performance Timing Analysis Tool for VLSI Systems.

[LLVM](#) is a library that has collection of modular/reusable compiler and toolchain components (assemblers, compilers, debuggers, etc.). With these components LLVM can be used as a compiler framework, providing a front-end(parser and lexer) and a back-end (code that converts LLVM's representation to actual machine code).

[TinyGo](#) is a Go compiler(based on LLVM) intended for use in small places such as microcontrollers, WebAssembly (Wasm), and command-line tools.

[Chipyard](#) is an open source framework for agile development of Chisel-based systems-on-chip. It will allow you to leverage the Chisel HDL, Rocket Chip SoC generator, and other [Berkeley](#) projects to produce a RISC-V SoC with everything from MMIO-mapped peripherals to custom accelerators.

The [Eclipse Embedded CDT](#) is a collection of plug-ins for Arm & RISC-V C/C++ developers. [Unicorn](#) is a lightweight, multi-platform, multi-architecture CPU emulator framework(ARM, AArch64, M68K, Mips, Sparc, X86) based on [QEMU](#).

[Keystone](#) is a lightweight multi-platform, multi-architecture(Arm, Arm64, Hexagon, Mips, PowerPC, Sparc, SystemZ & X86) assembler framework.

[Reko](#) is a decompiler for machine code binaries.

[Renode](#) is [Antmicro's](#) virtual development framework for multinode embedded networks (both wired and wireless) and is intended to enable a scalable workflow for creating effective, tested and secure IoT systems.

[Diosix](#) is a lightweight, secure, multiprocessor bare-metal hypervisor written in Rust for RISC-V.

70. Wayland

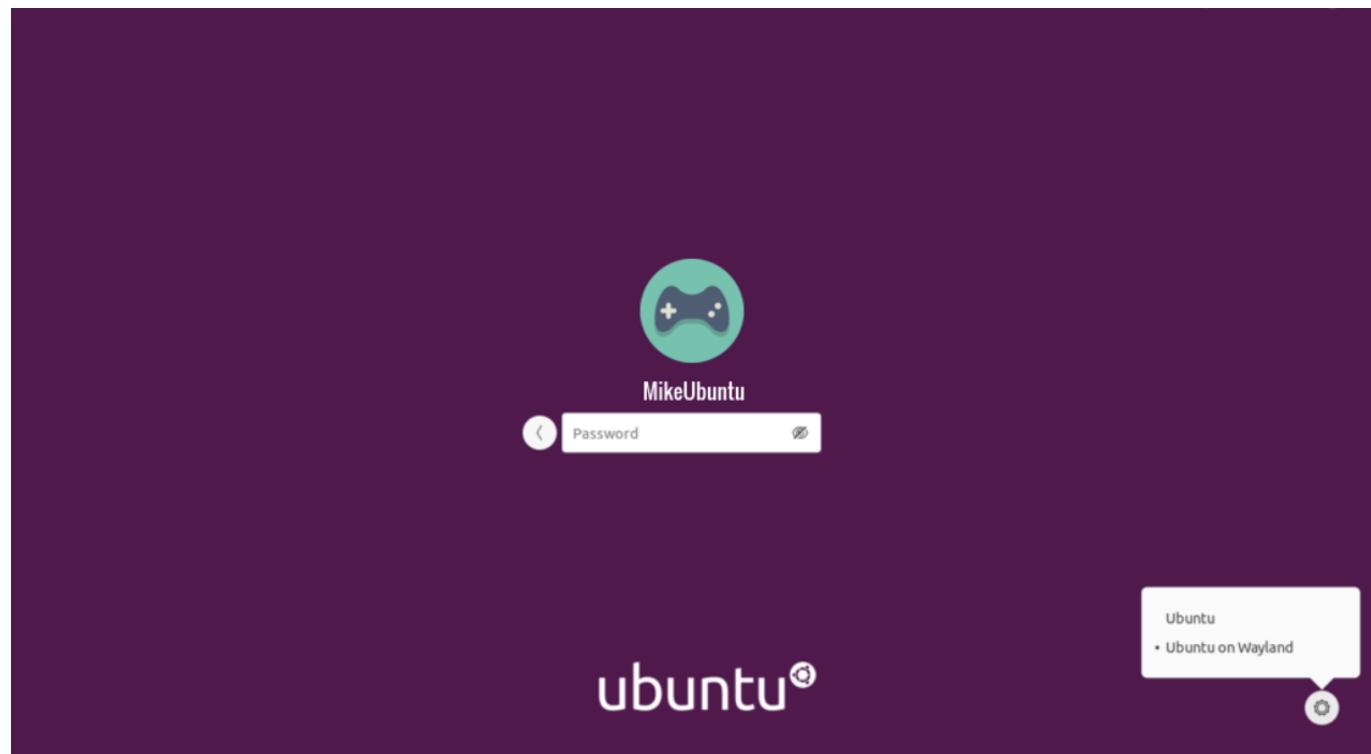
[Back to the Top](#)



Wayland Guide

A guide covering Wayland a small display server protocol and IPC(Inter-process communication) library. This includes the applications, libraries and tools that will make you a better and more efficient developer with Wayland.

Running Wayland on Ubuntu 20.04



Note: Wayland does run by default on the [Fedora](#) desktop.

Resources

[Wayland](#) is a protocol for a compositor to talk to its clients as well as a C library implementation of that protocol. The compositor can be a standalone display server running on Linux kernel modesetting and

evdev input devices, an [X application](#), or a wayland client itself.

[Wayland Architecture](#)

[Wayland Documentation](#)

Sotfware Toolkits that have Wayland support right now

[Contribution instructions for Wayland](#)

[Contribution instructions for Weston](#)

[Reporting Wayland bugs](#)

[Reporting Weston bugs](#)

[WSLG: X11 and Wayland Applications in Windows Subsystem for Linux\(WSL2\)](#)

[Qt Wayland Compositor](#)

[Qt Wayland Compositor Examples](#)

[Wayland on ArchWiki](#)

[Sway on ArchWiki](#)

[Wayland on Ubuntu Wiki](#)

[Wayland on Debian Wiki](#)

[The Wayland Display Server on Fedora Docs](#)

[Wayland features on Fedora Project Wiki](#)

[Wayland on GNOME Wiki](#)

[KWin/Wayland on KDE Community Wiki](#)

[Wayland Desktop Landscape on Gentoo Wiki](#)

[Wayland in Void Linux Handbook](#)

[Wayland on Enlightenment DE](#)

Tools

[Weston](#) is a lightweight and functional Wayland compositor.

[XWayland](#) is an X Server running as a Wayland client(for backwards compatibility), allowing the [Xorg server](#) can be modified to use wayland input devices for input and forward either the root window or individual top-level windows as wayland surfaces.

[KWayland](#) is a Qt-style API to interact with the wayland-client and wayland-server API.

[Qt](#) is the faster, smarter way to create innovative devices, modern UIs & applications for multiple screens. It is one of the most popular toolkits for the Wayland and X11 windowing.

[GTK](#) is a free and open source cross-platform widget toolkit for creating graphical user interfaces developed by [GNOME Project](#). It is one of the most popular toolkits for the Wayland and X11 windowing.

[NVIDIA Wayland EGL External Platform library](#) is a work-in-progress implementation of a EGL External Platform library to add client-side Wayland support to EGL on top of EGLDevice and EGLStream families of extensions.

[NVIDIA EGL External Platform Interface](#) is a work-in-progress specification of the EGL External Platform interface for writing EGL platforms and their interactions with modern window systems on top of existing low-level EGL platform implementations. This keeps window system implementation specifics out of EGL drivers by using application-facing EGL functions.

[Sway](#) is an i3-compatible Wayland compositor.

[wlroots](#) is a modular Wayland compositor library.

[WayfireWM](#) is a 3D Wayland compositor, inspired by [Compiz](#) and based on [wlroots](#).

[SDDM](#) is a modern display manager for X11 and Wayland aiming to be fast, simple and beautiful. It uses modern technologies like QtQuick, which in turn gives the designer the ability to create smooth, animated user interfaces.

[x11docker](#) is an application that you allows to run graphical desktop applications (and entire desktops) in Docker Linux containers.

[Mako](#) is alightweight notification daemon for Wayland. It also works on [Sway](#).

[Wayland-rs](#) is a Rust implementation of the wayland protocol (client and server).

[Wine-wayland](#) is an application that allows you to running DX9/DX11 and Vulkan games using pure Wayland and Wine/DXVK.

71. Electron

[Back to the Top](#)



Electron Guide

A guide covering the Electron framework including the applications, libraries and tools that will make you a better and more efficient developer with Electron.

Resources

[Electron](#) is a framework lets you write cross-platform desktop applications using JavaScript, HTML and CSS. It is based on [Node.js](#) and [Chromium](#) and is used by the [Atom editor](#) and many other [apps](#).

The [OpenJS Foundation](#) is made up of 32 open source JavaScript projects including Appium, Dojo, Electron, jQuery, Node.js, and webpack. The foundation's mission is to support the healthy growth of JavaScript and web technologies by providing a neutral organization to host and sustain projects, as well as collaboratively fund activities that benefit the ecosystem as a whole.

Electron Apps

[Getting Started with Electron](#)

[Electron Development](#)

[Configuring JavaScript libraries in WebStorm](#)

[JavaScript in Visual Studio Code](#)

[JavaScript extensions for VS Code](#)

[Master Electron: Desktop Apps with HTML, JavaScript & CSS course on Udemy](#)

[Electron for Desktop Apps: The Complete Developer's Guide course on Udemy](#)

[Electron From Scratch: Build Desktop Apps With JavaScript course on Udemy](#)

[Electron Courses on Coursera](#)

[Electron Fundamentals on Pluralsight](#)

Tools

[Electron Fiddle](#) is an application that lets you create and play with small Electron experiments. It greets you with a simple quick start template after opening. Simply choose the version of Electron you want to run with your project and then play around.

[Electron Builder](#) is a complete solution to package and build a ready for distribution Electron app with "auto update" support out of the box.

[Photon](#) is a UI toolkit for building desktop apps with Electron.

[Electron.NET](#) is an application that builds cross platform desktop apps with ASP.NET Core (Razor Pages, MVC, Blazor).

[Angular Electron](#) is an application that bootstrap's and package's your project with Angular 11 and Electron 11 (Typescript + SASS + Hot Reload) for creating Desktop applications.

[Selenium](#) is a browser automation framework and ecosystem. Selenium specifically provides an infrastructure for the [W3C WebDriver specification](#) as a platform and language-neutral coding interface compatible with all major web browsers(Firefox, Google Chrome and Safari).

[Selenium IDE](#) is an Open Source record and playback test automation for the web.

[GitHub Codespaces](#) is an integrated development environment(IDE) on GitHub. That allows developers to develop entirely in the cloud using Visual Studio and Visual Studio Code.

[GitHub Actions](#) will automate, customize, and execute your software development workflows right in your repository with GitHub Actions. You can discover, create, and share actions to perform any job you'd like, including CI/CD, and combine actions in a completely customized workflow.[GitHub Actions for Azure](#) you can create workflows that you can set up in your repository to build, test, package, release and deploy to Azure.Learn more about all other integrations with Azure.

[WebStorm](#) is a professional IDE for JavaScript(including support for both HTML and CSS) developed by JetBrains. WebStorm comes with intelligent code completion, on-the-fly error detection, powerful navigation and refactoring for JavaScript, TypeScript, stylesheet languages, and all the most popular frameworks([Angular](#), [React](#), [Vue.js](#), [Ionic](#), [Apache Cordova](#), [React Native](#), [Node.js](#), [Meteor](#), and [Electron](#)).

[Visual Studio Code](#) is a code editor redefined and optimized for building and debugging modern web and cloud applications.

[GraphQL](#) is a query language for APIs and a runtime for fulfilling those queries with your existing data. It has support in Java, JavaScript, Ruby, Scala, and other programming languages.

[TypeORM](#) is an ORM that can run in NodeJS, Browser, Cordova, PhoneGap, Ionic, React Native, NativeScript, Expo, and Electron platforms and can be used with TypeScript and JavaScript (ES5, ES6, ES7, ES8).

[Nest](#) is a framework for building efficient, scalable [Node.js](#) server-side applications. It uses modern JavaScript, is built with TypeScript (preserves compatibility with pure JavaScript) and combines elements of OOP (Object Oriented Programming), FP (Functional Programming), and FRP (Functional Reactive Programming).

[Quasar](#) is a framework that builds high-performance VueJS user interfaces in record time. Such as responsive Single Page Apps, SSR Apps, PWAs, Browser extensions, Hybrid Mobile Apps and Electron Apps.

[Gatsby](#) is a free and open source framework based on React that helps developers build blazing fast websites and apps.

[React Starter Kit](#) is an isomorphic web app boilerplate for web development built on top of [Node.js](#), [Express](#), [GraphQL](#) and [React](#), containing modern web development tools such as [Webpack](#), [Babel](#) and [Browsersync](#). Helping you to stay productive following the best practices.

[Enzyme](#) is a JavaScript Testing utility for React that makes it easier to test your React Components' output. The user can also manipulate, traverse, and in some ways simulate runtime given the output.

[Mocha](#) is a simple, flexible, fun JavaScript test framework for Node.js & The Browser.

[Express](#) is a fast, unopinionated, minimalist web framework for node.

[Meteor](#) is an ultra-simple environment for building modern web applications with JavaScript.

[RxDB](#) is a NoSQL-database for JavaScript Applications like Websites, hybrid Apps, Electron-Apps, Progressive Web Apps and NodeJs.

[Redux](#) is a predictable state container for JavaScript apps.

[Standard](#) is a JavaScript Style Guide, with linter & automatic code fixer.

[Lowdb](#) is a small local JSON database powered by Lodash (supports Node, Electron and the browser).

[Nativefier](#) is an application that lets you make any web page into a desktop electron application.

72. jQuery

[Back to the Top](#)



jQuery Guide

A guide covering jQuery(JavaScript library) including the applications, libraries and tools that will make you a better and more efficient developer with jQuery.

Resources

[The OpenJS Foundation](#) is made up of 32 open source JavaScript projects including Appium, Dojo, Electron, jQuery, Node.js, and webpack. The foundation's mission is to support the healthy growth of JavaScript and web technologies by providing a neutral organization to host and sustain projects, as well as collaboratively fund activities that benefit the ecosystem as a whole.

[jQuery](#) is a fast and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of web browsers.

[jQuery API Documentation](#)

[jQuery Learning Center](#)

[Configuring JavaScript libraries in WebStorm](#)

[JavaScript in Visual Studio Code](#)

[JavaScript extensions for VS Code](#)

[jQuery Courses on Coursera](#)

[jQuery Courses on Udemy](#)

[jQuery Training Courses on LinkedIn Learning](#)

[Intro to jQuery on Udacity](#)

[Intro to AJAX on Udacity](#)

[Learning jQuery on Codecademy](#)

[jQuery Best Practices Training on Learning Tree](#)

[Learning the jQuery Basics on Pluralsight](#)

[jQuery In-Depth on Pluralsight](#)

[ASP.NET Ajax JavaScript and jQuery Course on Pluralsight](#)

[jQuery Certification from W3Schools](#)

Tools

[WebStorm](#) is a professional IDE for JavaScript(including support for both HTML and CSS) developed by JetBrains. WebStorm comes with intelligent code completion, on-the-fly error detection, powerful navigation and refactoring for JavaScript, TypeScript, stylesheet languages, and all the most popular frameworks([Angular](#), [React](#), [Vue.js](#), [Ionic](#), [Apache Cordova](#), [React Native](#), [Node.js](#), [Meteor](#), and [Electron](#)).

[Visual Studio Code](#) is a code editor redefined and optimized for building and debugging modern web and cloud applications.

[GraphQL](#) is a query language for APIs and a runtime for fulfilling those queries with your existing data. It has support in Java, JavaScript, Ruby, Scala, and other programming languages.

[TypeORM](#) is an ORM that can run in NodeJS, Browser, Cordova, PhoneGap, Ionic, React Native, NativeScript, Expo, and Electron platforms and can be used with TypeScript and JavaScript (ES5, ES6, ES7, ES8).

[Cheerio](#) is a fast, flexible, and lean implementation of core jQuery designed specifically for the server.

[jQuery UI](#) is a curated set of user interface interactions, effects, widgets, and themes built on top of jQuery. Though, it should not be used in core code.

[jQuery Tools](#) is a collection of the most important user-interface components for modern websites. Used by large sites all over the world.

[Bridge.NET](#) is an open source C#-to-JavaScript Compiler. Write your application in C# and run on billions of devices.

[Phantomas](#) is a Headless Chromium-based web performance metrics collector and monitoring tool

[Featherlight](#) is a very lightweight jQuery lightbox plugin. It's simple yet flexible and easy to use.

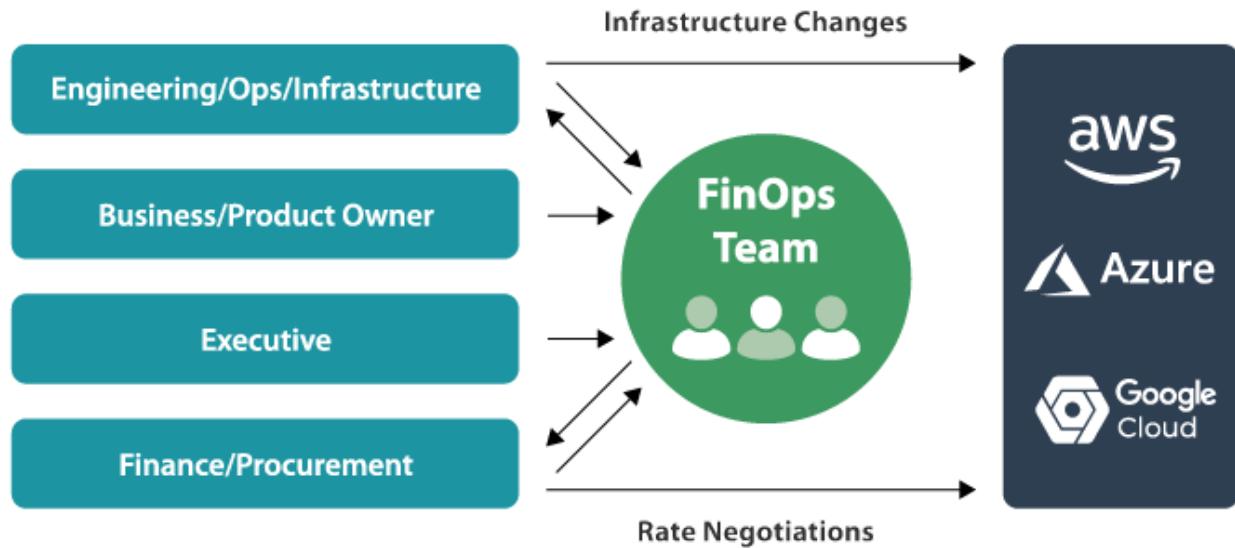
73. FinOps

[Back to the Top](#)



FinOps Guide

A guide covering FinOps including the applications, tools and training that will make your development team better and more efficient at Cloud Financial Management.



FinOps Team Structure. Source: [FinOps Foundation, 2020](#).

Resources

[FinOps\(finance operations\)](#) is the practice of bringing financial accountability to the variable spend model of cloud, enabling distributed teams to make business trade-offs between speed, cost, and quality. At its core, FinOps is a cultural practice for Cloud Financial Management. The [FinOps Foundation joins the Linux Foundation in August 2020](#).

[Linux Foundation](#) is home to Linux, Node.js and other mission critical projects that form the backbone of modern internet services, including Pinterest. Joining The Linux Foundation is a great way for established companies like ours to support those communities.

[FinOps Landscape](#)

[FinOps Foundation Principles](#)

[FinOps Technical Advisory Council \(TAC\)](#)

[The FinOps Ambassadors program](#)

[Introduction to FinOps](#)

[FinOps Training & Certification](#)

[FinOps Certified Platform](#)

[FinOps Certified Service Provider](#)

[FinOps Training Partner](#)

[FinOps Foundation Training Events](#)

[FinOps at your Organization course on Udemy](#)

[FinOps Foundation GitHub](#)

[Rate Optimization: Google Cloud CUD & Big Query Strategy](#)

[AWS Savings Plans and FinOps](#)

[Introducing FinOps fro AWS Cloud](#)

[Using FinOps with AWS by Apptio Cloudability](#)

[Cloud Economics v2.0 - Part 1: FinOps](#)

Tools

[Dynamics 365 Finance and Operations Tools](#) is an extension for Visual Studio provides Azure DevOps build and release tasks for Dynamics 365 Finance and Operations and Dynamics Lifecycle Services.

[Visual Studio](#) is an integrated development environment (IDE) from Microsoft; which is a feature-rich application that can be used for many aspects of software development. Visual Studio makes it easy to edit, debug, build, and publish your app. By using Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, and Windows Store.

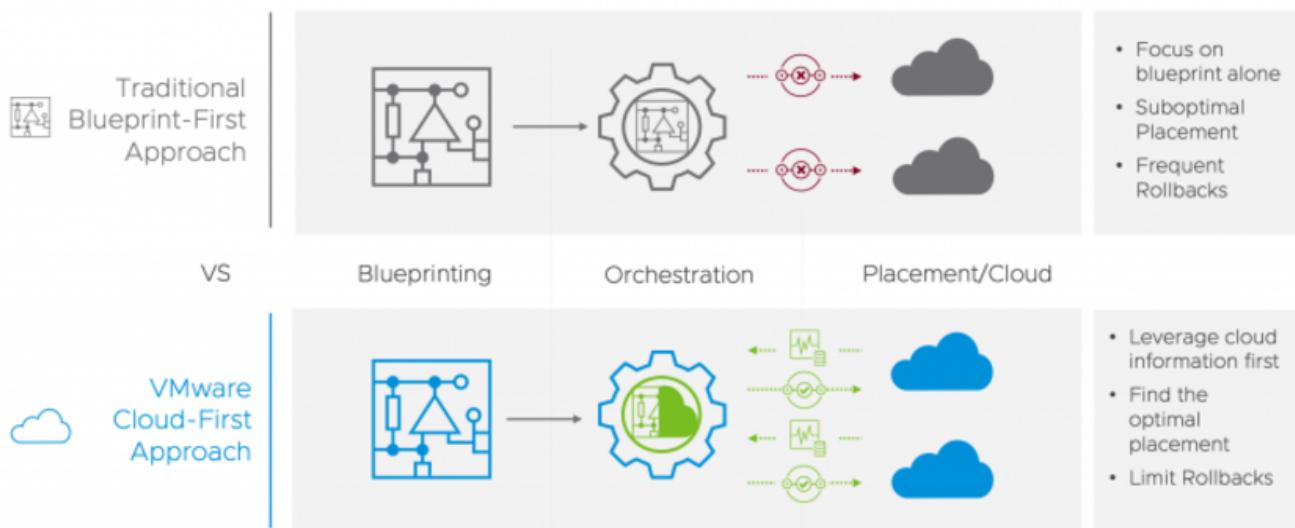
74. Telco 5G

[Back to the Top](#)

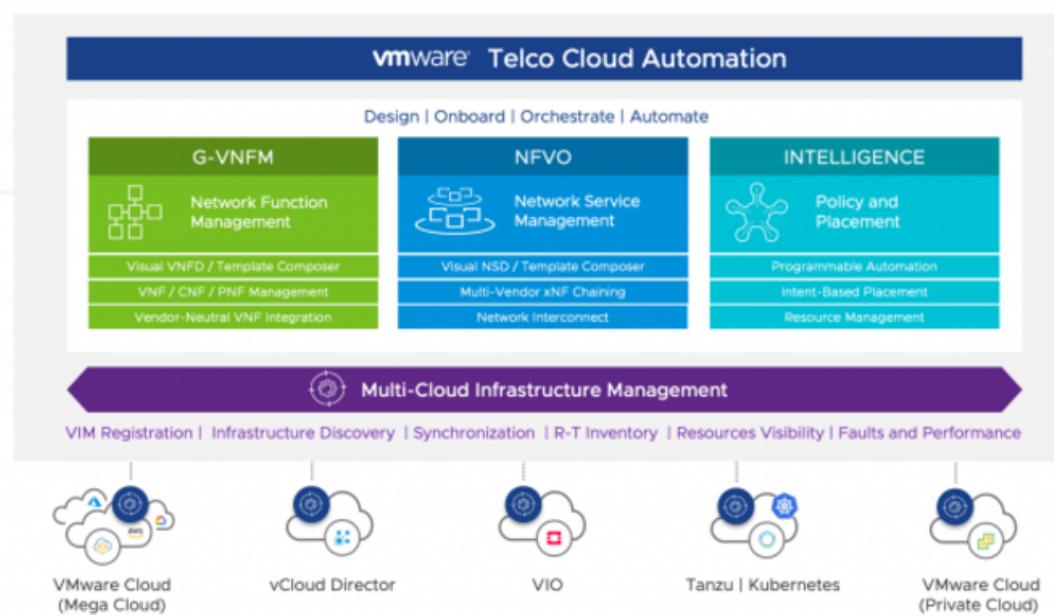


Telco 5G Guide

A guide covering Telco 5G including the applications, tools and training that will make your development team better and more efficient at implementing Telco 5G networking solutions.



VMware Cloud First Approach. Source: [VMware, 2020](#).



VMware Telco Cloud Automation Components. Source: [VMware, 2020](#).

Resources

[HPE\(Hewlett Packard Enterprise\) Telco Blueprints overview](#)

[Network Functions Virtualization Infrastructure \(NFVI\) by Cisco](#)

[Introduction to vCloud NFV Telco Edge from VMware](#)

[VMware Telco Cloud Automation\(TCA\) Architecture Overview](#)

[5G Telco Cloud from VMware](#)

[Maturing OpenStack Together To Solve Telco Needs from Red Hat](#)

[Red Hat telco ecosystem program](#)

[OpenStack for Telcos by Canonical](#)

[Open source NFV platform for 5G from Ubuntu](#)

[Understanding 5G Technology from Verizon](#)

[Verizon and Unity partner to enable 5G & MEC gaming and enterprise applications](#)

[Understanding 5G Technology from Intel](#)

[Understanding 5G Technology from Qualcomm](#)

[Telco Acceleration with Xilinx](#)

[VIMs on OSM Public Wiki](#)

[Amazon EC2 Overview and Networking Introduction for Telecom Companies](#)

[Citrix Certified Associate – Networking\(CCA-N\)](#)

[Citrix Certified Professional – Virtualization\(CCP-V\)](#)

[CCNP Routing and Switching](#)

[Certified Information Security Manager\(CISM\)](#)

[Wireshark Certified Network Analyst \(WCNA\)](#)

[Juniper Networks Certification Program Enterprise \(JNCP\)](#)

[Cloud Native Computing Foundation Training and Certification Program](#)

Tools

[Open Stack](#) is an open source cloud platform, deployed as infrastructure-as-a-service (IaaS) to orchestrate data center operations on bare metal, private cloud hardware, public cloud resources, or both (hybrid/multi-cloud architecture). OpenStack includes advance use of virtualization & SDN for network traffic optimization to handle the core cloud-computing services of compute, networking, storage, identity, and image services.

[StarlingX](#) is a complete cloud infrastructure software stack for the edge used by the most demanding applications in industrial IOT, telecom, video delivery and other ultra-low latency use cases.

[Airship](#) is a collection of open source tools for automating cloud provisioning and management. Airship provides a declarative framework for defining and managing the life cycle of open infrastructure tools and the underlying hardware.

[Network functions virtualization \(NFV\)](#) is the replacement of network appliance hardware with virtual machines. The virtual machines use a hypervisor to run networking software and processes such as routing and load balancing. NFV allows for the separation of communication services from dedicated hardware, such as routers and firewalls. This separation means network operations can provide new services dynamically and without installing new hardware. Deploying network components with network functions virtualization only takes hours compared to months like with traditional networking solutions.

Software Defined Networking (SDN) is an approach to networking that uses software-based controllers or application programming interfaces (APIs) to communicate with underlying hardware infrastructure and direct traffic on a network. This model differs from that of traditional networks, which use dedicated hardware devices (routers and switches) to control network traffic.

Virtualized Infrastructure Manager (VIM) is a service delivery and reduce costs with high performance lifecycle management Manage the full lifecycle of the software and hardware comprising your NFV infrastructure (NFVI), and maintaining a live inventory and allocation plan of both physical and virtual resources.

Management and Orchestration(MANO) is an ETSI-hosted initiative to develop an Open Source NFV Management and Orchestration (MANO) software stack aligned with ETSI NFV. Two of the key components of the ETSI NFV architectural framework are the NFV Orchestrator and VNF Manager, known as NFV MANO.

Magma is an open source software platform that gives network operators an open, flexible and extendable mobile core network solution. Their mission is to connect the world to a faster network by enabling service providers to build cost-effective and extensible carrier-grade networks. Magma is 3GPP generation (2G, 3G, 4G or upcoming 5G networks) and access network agnostic (cellular or WiFi). It can flexibly support a radio access network with minimal development and deployment effort.

OpenRAN is an intelligent Radio Access Network(RAN) integrated on general purpose platforms with open interface between software defined functions. Open RANecosystem enables enormous flexibility and interoperability with a complete openness to multi-vendor deployments.

Open vSwitch(OVS) is an open source production quality, multilayer virtual switch licensed under the open source Apache 2.0 license. It is designed to enable massive network automation through programmatic extension, while still supporting standard management interfaces and protocols (NetFlow, sFlow, IPFIX, RSPAN, CLI, LACP, 802.1ag).

Edge is a distributed computing framework that brings enterprise applications closer to data sources such as IoT devices or local edge servers. This proximity to data at its source can deliver strong business benefits, including faster insights, improved response times and better bandwidth availability.

Multi-access edge computing (MEC) is an Industry Specification Group (ISG) within ETSI to create a standardized, open environment which will allow the efficient and seamless integration of applications from vendors, service providers, and third-parties across multi-vendor Multi-access Edge Computing platforms.

Virtualized network functions(VNFs) is a software application used in a Network Functions Virtualization (NFV) implementation that has well defined interfaces, and provides one or more component networking functions in a defined way. For example, a security VNF provides Network Address Translation (NAT) and firewall component functions.

Cloud-Native Network Functions(CNF) is a network function designed and implemented to run inside containers. CNFs inherit all the cloud native architectural and operational principles including Kubernetes(K8s) lifecycle management, agility, resilience, and observability.

Physical Network Function(PNF) is a physical network node which has not undergone virtualization. Both PNFs and VNFs (Virtualized Network Functions) can be used to form an overall Network Service.

Network functions virtualization infrastructure(NFVI) is the foundation of the overall NFV architecture. It provides the physical compute, storage, and networking hardware that hosts the VNFs. Each NFVI block can be thought of as an NFVI node and many nodes can be deployed and controlled geographically.

Contribute

[Back to the Top](#)

- If would you like to contribute to this handbook simply make a [Pull Request](#).

License

[Back to the Top](#)

Distributed under the [Creative Commons Attribution 4.0 International \(CC BY 4.0\) Public License](#).