3.26.2 - Latest

Filter by keyword

Guides - Latest

Tutorials

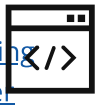
Short and focused exercises to get you going quickly.

[Building a Native Executable](#)

Build native executables with GraalVM or Mandrel.

[Collect metrics using Micrometer](#)

Create an application that uses the Micrometer metrics library to collect runtime, extension and application metrics and expose them as a Prometheus (OpenMetrics) endpoint.



[Creating Your First Application](#)

Discover how to create your first Quarkus application.

[Creating a tutorial](#)

Create a new tutorial that guides users through creating, running, and testing a Quarkus application that uses annotations from an imaginary extension.

[Getting Started With Reactive](#)

Learn more about developing reactive applications with Quarkus.

[Getting started with WebSockets Next](#)

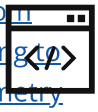
This guide explains how your Quarkus application can utilize web sockets to create interactive web applications.

[Getting started with security by using Basic authentication and Jakarta Persistence](#)

Get started with Quarkus Security by securing your Quarkus application endpoints with the built-in Quarkus Basic authentication and the Jakarta Persistence identity

[Migrate from OpenTracing to OpenTelemetry tracing](#)

Migrate an application from OpenTracing to OpenTelemetry tracing in Quarkus 3.x.



[Migrate from Vert.x OIDC to Quarkus OIDC](#)

Learn how to migrate your Vert.x OIDC application to Quarkus and choose to either retain Vert.x OIDC or replace it with

provider, enabling role-based access control.

Quarkus OIDC.

[OpenID Connect client and token propagation quickstart]

Learn how to use OpenID Connect (OIDC) and OAuth2 clients with filters to get, refresh, and propagate access tokens in your applications.

[Protect Quarkus web application by using an Auth0 OpenID Connect provider]

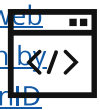
Quarkus Security provides comprehensive OpenId Connect (OIDC) and OAuth2 support with its quarkus-oidc extension, supporting both Authorization code flow and Bearer token authentication mechanisms.

oidc, sso, auth0

[Protect a service application by using OpenID Connect (OIDC) Bearer token authentication]

Use the Quarkus OpenID Connect (OIDC) extension to secure a Jakarta REST application with Bearer token authentication.

[Protect a web application by using OpenID Connect (OIDC) authorization code flow]



[Quarkus Tools in your favorite IDE]

Learn more about Quarkus

[Using our Tooling]

Explore the Quarkus developer toolchain which makes Quarkus

Discover how to secure application HTTP endpoints by using the Quarkus OpenID Connect (OIDC) authorization code flow mechanism with the Quarkus OIDC extension, providing robust authentication and authorization.

integrations in IDEs.

development so fast and enjoyable.

[Your second Quarkus application]

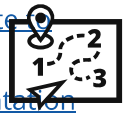
Discover some of the features that make developing with Quarkus a joyful experience.

How-to Guides

Step-by-step guides to covering key tasks, real world operations and common problems.

Contribute to Quarkus documentation

Contribute to the documentation by using the recommended diataxis content types, steps, workflow, and style guidance to ensure the content successfully renders on the Quarkus website portal.



Deploying Quarkus Java applications to OpenShift by using a Docker build strategy

This guide describes how to build and deploy a Quarkus application on OpenShift by using the Docker build strategy.

Deploying Quarkus applications compiled to native executables

This guide describes how to deploy a Quarkus application to OpenShift compiled to native executables.

Deploying Quarkus applications to OpenShift in a single step

This guide describes how to build and deploy a Quarkus application to OpenShift in a single step.

Dev Services and Dev UI for OpenID Connect (OIDC)

You can use Dev Services for Keycloak and the Dev UI for the OpenID Connect (OIDC)

Enable Basic authentication

Enable Basic authentication for your Quarkus project and allow users to authenticate with a username and password.

Keycloak provider and adapt these services for other OpenID Connect providers.

sso oidc security keycloak

[Frequently asked questions about writing extensions]

[Quarkus Security with Jakarta Persistence]

You can configure your application to use Jakarta Persistence to store users' identities.

[Update projects to the latest Quarkus version]

Learn how to upgrade your projects to the latest version of Quarkus

[Use virtual threads in REST applications]

How to use virtual threads in a REST application

[Using OpenID Connect (OIDC) and Keycloak to centralize authorization]

Learn how to enable

[Using OpenID Connect (OIDC) multitenancy]

This guide demonstrates how your OpenID Connect

bearer token authorization in your Quarkus application by using Keycloak Authorization Services for secure access to protected resources.

sso, oidc, security, keycloak

(OIDC) application can support multitenancy to serve multiple tenants from a single application.

sso oidc oauth2 security

[Using S2I to deploy Quarkus applications to OpenShift]

This guide describes how to build and deploy a Quarkus application on OpenShift by using Source-to-Image (S2I).

[Writing a Dev Service]

Learn how to develop a Dev Service for your extension in order to replace an external service in development mode.

[YAML configuration]

Optionally, use application.yaml instead of application.properties to configure your application.

Concepts

Explanations of some of the larger concepts and technologies involved with Quarkus.

A maturity matrix for Quarkus extensions

Quarkus extensions can do a lot, or a little.

Authentication mechanisms in Quarkus

The Quarkus Security framework supports multiple authentication mechanisms, which you can use to secure your applications.

Basic authentication

HTTP Basic authentication is one of the least resource-demanding techniques that enforce access controls to web resources.

Configuring Well-Known OpenID Connect Providers

This document explains how to configure well-known social OIDC and OAuth2 providers.

oidc github twitter google facebook mastodon

Duplicated context, context locals, asynchronous processing and propagation

When using a traditional, blocking, and synchronous framework, processing of each request is performed in a dedicated thread.

Identity providers

In the Quarkus Security framework, identity providers play a crucial role in authentication and authorization by verifying user identities.

microsoft
apple
spotify
twitch
linkedin
strava

OpenID Connect (OIDC) Bearer token authentication

Secure HTTP access to Jakarta REST (formerly known as JAX-RS) endpoints in your application with Bearer token authentication by using the Quarkus OpenID Connect (OIDC) extension.

OpenID Connect authorization code flow mechanism for protecting web applications

To protect your web applications, you can use the industry-standard OpenID Connect (OIDC) Authorization Code Flow mechanism provided by the Quarkus OIDC extension.

Proactive authentication

Learn how to manage proactive authentication in Quarkus, including customizing settings and handling exceptions.

Quarkus OpenId Connect (OIDC) Expanded Configuration Reference

Quarkus OIDC quarkus-oidc extension

Quarkus Security architecture

The Quarkus Security architecture provides several built-

Quarkus Security overview

Quarkus Security is a framework that provides the

provides a comprehensive, highly adaptable and configurable OIDC and OAuth2 adapter implementation.

in authentication mechanisms and is highly customizable.

architecture, multiple authentication and authorization mechanisms, and other tools to build secure and production-quality Java applications.

Quarkus documentation content types

Quarkus documentation is structured into four distinct content types: concepts, how-tos, tutorials, and references.

Security vulnerability detection and reporting in Quarkus

Most of the Quarkus tags are registered in the US National Vulnerability Database (NVD) in Common Platform Enumeration (CPE) name format.

References

Technical Resource that covers tools, components, and commands. The encyclopedia for Quarkus.

[Authorization of web endpoints]

Quarkus incorporates a pluggable web security layer.

[Class Loading Reference]

Learn more about Quarkus class loading infrastructure.

[Command Mode Applications]

This reference guide explains how to develop command line applications with Quarkus.

[Configuration Reference Guide]

Learn more about how to configure your Quarkus applications.

[Configure data sources in Quarkus]

Use a unified configuration model to define data sources for Java Database Connectivity (JDBC) and Reactive drivers.

[Contexts and Dependency Injection]

Go more in depth into the Quarkus implementation of CDI.

arc

[Cross-Origin Resource Sharing (CORS)]

Enable and configure CORS in Quarkus to specify

[HTTP Reference]

Learn more about configuring Quarkus' Vert.x based

[Infinispan Cache]

Use Infinispan as the Quarkus cache backend

allowed origins, methods, and headers, guiding browsers in handling cross-origin requests safely.

cors, http, configuration, security, headers

HTTP layer - and Undertow if you are using servlets.

[Infinispan Client Extension Reference Guide](#)

Infinispan is an in memory distributed data store and cache server that offers flexible deployment options and robust capabilities for storing, managing, and processing data.

[Load Shedding reference guide](#)

Load shedding is the practice of detecting service overload and rejecting requests.

[Logging configuration](#)

Read about the use of logging API in Quarkus, configuring logging output, and using logging adapters to unify the output from other logging APIs.

[Mailer Reference](#)

[Management interface](#)

[Micrometer Metrics](#)

Guide

This reference guide explains in more details the configuration and usage of the Quarkus Mailer.

reference

Management interface configuration

Use Micrometer to collect metrics produced by Quarkus, its extensions, and your application.

Micrometer and OpenTelemetry extension

Guide to send Micrometer data to OpenTelemetry.

Native Reference Guide

This guide is a companion to the Building a Native Executable, Using SSL With Native Images, and Writing Native Applications, guides.

Observability in Quarkus

This guide explains how your Quarkus application can utilize OpenTelemetry to provide observability for interactive web applications.

OpenID Connect (OIDC) and OAuth2 client and filters

You can use Quarkus extensions for OpenID Connect and

OpenID Connect (OIDC) and OAuth2 dynamic client registration

Typically, you have to register an

OpenID Connect (OIDC) configurati properties

As a Quarkus developer, you confi the Quarkus OpenID Connect (OID extension by setting the following properties in the src/main/resources/application.pr file.

OAuth 2.0 access token management, focusing on acquiring, refreshing, and propagating tokens.

OIDC client (application) manually in your OIDC provider's dashboard.

Quarkus Maven Plugin

The Quarkus Maven Plugin builds the Quarkus applications, and provides helpers to launch dev mode or build native executables.

Quarkus style and content guidelines

Guidelines are provided to help you to contribute clear and consistent content that is also sourced in the required diataxis structure and composition of Quarkus documentation.

Qute Reference Guide

Learn everything you need to know about the Qute template engine.

Reactive Messaging AMQP 1.0 Connector Reference Documentation

This guide is the companion from the Getting Started with AMQP 1.0.

Reactive Messaging RabbitMQ Connector Reference Documentation

This guide is the companion from the Getting Started with RabbitMQ.

Redis Cache

Use Redis as the Quarkus cache backend

[Redis Extension Reference Guide](#)

Redis is an in-memory data store used as a database, cache, streaming engine, and message broker.

[Scheduler Reference Guide](#)

Learn more about the Scheduler extension.

[Stork Reference Guide](#)

This guide is the companion from the Stork Getting Started Guide.

[TLS registry reference](#)

TLS registry configuration and usage

[Using OpenTelemetry](#)

This guide explains how your Quarkus application can utilize OpenTelemetry to provide observability for interactive web applications.

[Using transactions in Quarkus](#)

The quarkus-narayana-jta extension provides a Transaction Manager that coordinates and expose transactions to your applications as described in the link: Jakarta Transactions specification, formerly known as

Java Transaction API (JTA).

Vert.x Reference Guide

This reference guide provides advanced details about the usage and the configuration of the Vert.x instance used by Quarkus.

vertx event verticle

Virtual Thread support reference

This guide explains how to benefit from Java 21+ virtual threads in Quarkus application.

WebSockets Next reference guide

The quarkus-websockets-next extension provides a modern declarative API to define WebSocket server and client endpoints.

gRPC code generation reference guide

Learn how to configure gRPC code generation.

gRPC reference guide

Learn how to configure gRPC server and clients.

General Guides

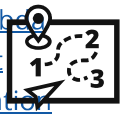
Other Quarkus Guides

[AWS Lambda]

This guide explains how you can deploy Quarkus-based AWS Lambdas.

[AWS Lambda SnapStart Configuration]

This document explains how to optimize your AWS Lambda application for SnapStart

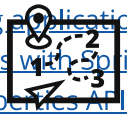


[AWS Lambda with Quarkus REST, Undertow, or Reactive Routes]

This guide explains how you can deploy Vert.x Web, Servlet, or RESTEasy microservices as an AWS Lambda.

[Accessing application properties with Spring Boot properties API]

Use Spring Boot's @ConfigurationProperties in place of MicroProfile Config annotations



[All configuration options]

List all the configuration properties per extensions

[Apache Camel on Quarkus]

This guide covers the systems integration with Apache Camel

[Apache Kafka Reference Guide]

This reference guide

[Apache Pulsar Reference Guide]

This reference guide

[AppCDS]

This reference guide explains how to enable

provides an in-depth look on Apache Kafka and Quarkus Messaging extensions.

provides an in-depth look on Apache Pulsar and the Quarkus Messaging extensions.

AppCDS with Quarkus.

Application Data Caching

This guide explains how to cache expensive method calls of your CDI beans using simple annotations.

Application Initialization and Termination

You often need to execute custom actions when the application starts and clean up everything when the application stops.

lifecycle event

Automate Quarkus deployment with Ansible

Build and deploy your Quarkus App using Ansible

Azure Functions

Integrate Quarkus with the Microsoft Azure functions that you

Azure Functions with Quarkus REST, Undertow, or Reactive Routes

Deploy Vert.x Web, Servlet, or RESTEasy

Build Items

Explore all the BuildItems you can consume/produce in your extensions.

have written.

microservices as a Microsoft Azure Function.

[Build analytics]

This guide presents what build analytics is and how to configure it.

[Build, sign and encrypt JSON Web Tokens]

JSON Web Token (JWT) is defined by the RFC 7519 specification as a compact, URL-safe means of representing claims.

[Building Quarkus apps with Quarkus Command Line Interface (CLI)]

Use the Quarkus CLI to create, build, run, and manage extensions for Quarkus projects.

[Building my first extension]

Learn step by step how to build a simple extension.

[CDI Integration Guide]

Learn how to integrate your extension with Quarkus' CDI container.

[Centralized log management (Graylog, Logstash, Fluentd)]

This guide explains how to centralize your logs with Graylog, Logstash or Fluentd.

[Command Mode with Picocli]

Simplify command line applications creation with the Picocli extension.

[Compose Dev Services]

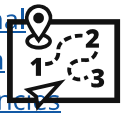
Configure custom Dev Services using Docker or Podman Compose.

[Compressing native executables using UPX]

Ultimate Packer for eXecutables (UPX) is a compression tool reducing the size of executables.

[Conditional Extension Dependencies]

Trigger the inclusion on additional extensions based on certain conditions.



[Configuring Your Application]

Hardcoded values in your code is a no go (even if we all did it at some point ;-)).

[Connecting to an Elasticsearch cluster]

This guide covers how to interact with an Elasticsearch cluster using the low level REST client or the Elasticsearch Java client.

[Consuming a gRPC Service]

This guide explains how to consume

[Container Images]

Learn how to build and push container images

[Context Propagation in Quarkus]

Learn more about how you can pass

gRPC services in your Quarkus application.

with Jib, OpenShift, Docker, or Podman as part of the Quarkus build.

contextual information with SmallRye Context Propagation.

[Continuous Testing]

Get early test feedback with Continuous Testing.

[Cross-Site Request Forgery Prevention]

Cross-Site Request Forgery (CSRF) is an attack that forces an end user to execute unwanted actions on a web application in which they are currently authenticated.

[Defining and executing business rules with Drools]

Drools is the most used rule engine implementation in the Java ecosystem.

[Deploying Quarkus applications to OpenShift]

This guide describes the deployment of Quarkus applications to OpenShift.

[Deploying to Google Cloud Platform (GCP)]

This guide explains how to deploy a Quarkus application to Google Cloud.

[Deploying to Heroku]

Deploy your Quarkus applications on Heroku.

[Deploying to Microsoft Azure Cloud]

Deploy a Quarkus application to the Microsoft Azure cloud platform.

[Deploying your gRPC Service in Kubernetes]

This guide explains how to deploy your gRPC services in Quarkus to Kubernetes.

[Dev Assistant]

Learn more about the Dev Assistant

[Dev MCP]

Learn more about Dev MCP

[Dev Services Overview]

An introduction to Dev Services and a list of all extensions that support Dev Services and their configuration options.

[Dev Services for AMQP]

Start AMQP automatically in dev and test modes.

[Dev Services for Apicurio Registry]

Start Apicurio Registry automatically

[Dev Services for Databases]

When testing or running in dev mode

[Dev Services for Elasticsearch]

Start Elasticsearch automatically in dev and test modes

in dev and test modes.

Quarkus can provide you with a zero-config database out of the box, a feature we refer to as Dev Services.

Dev Services for Infinispan

Start Infinispan automatically in dev and test modes.

Dev Services for Kafka

Start Apache Kafka automatically in dev and test modes.

Dev Services for Kubernetes

Start a Kubernetes API server automatically in dev and test modes.

Dev Services for MongoDB

Quarkus supports a feature called Dev Services that allows you to create various datasources without any config.

Dev Services for Pulsar

With Quarkus Messaging Pulsar extension (quarkus-messaging-pulsar) Dev Services for Pulsar automatically starts a Pulsar broker in dev

Dev Services for RabbitMQ

Dev Services for RabbitMQ automatically starts a RabbitMQ broker in dev mode and when running tests.

mode and when running tests.

[Dev Services for Redis](#)

Start Redis automatically in dev and test modes.

[Dev UI](#)

Learn more about Dev UI

[Extending Configuration Support](#)

Extend and customize the Configuration.

[Extension Capabilities](#)

How capabilities are implemented and used in Quarkus.

[Extension codestart](#)

Provide users with initial code for extensions when generating Quarkus applications on code.quarkus.io and all the Quarkus tooling.

[Extension for Spring Data API](#)

While you are encouraged to use Hibernate ORM with Panache for your data layer, Quarkus provides a compatibility layer for Spring Data JPA in the form of the spring-data-jpa extension.

[Extension for Spring Data REST](#)

[Funqy](#)

[Funqy AWS](#)

Spring Data REST simplifies the creation of CRUD applications based on our Spring Data compatibility layer.

This guide explains basics of the Funqy framework, a simple portable cross-provider cloud function API.

Lambda Binding

This guide explains Funqy's AWS Lambda binding.

Funqy Google Cloud Functions

This guide explains Funqy's Google Cloud Platform Functions binding.

Funqy HTTP Binding (Standalone)

This guide explains Funqy's HTTP binding.

Funqy HTTP Binding with AWS Lambda

This guide explains Funqy's AWS Lambda HTTP binding.

Funqy HTTP Binding with Azure Functions

Use Funqy HTTP binding with Microsoft Azure Functions to deploy your serverless

Funqy HTTP Binding with Google Cloud Functions

This guide explains Funqy's Google Cloud

Funqy Knative Events Binding

This guide explains Funqy's Knative Events binding.

Quarkus applications.

Platform Functions HTTP binding.

[Generating CycloneDX BOMs]

This guide explains how to generate SBOMs for Quarkus applications in the CycloneDX format.

[Generating Jakarta REST resources with Panache]

Hibernate ORM REST Data with Panache simplifies the creation of CRUD applications based on Jakarta REST and Hibernate ORM.

[Getting Started to Quarkus Messaging with AMQP 1.0]

This guide demonstrates how your Quarkus application can utilize Quarkus Messaging to interact with AMQP.

[Getting Started to Quarkus Messaging with Apache Kafka]

This guide demonstrates how your Quarkus application can utilize Quarkus Messaging to interact with

[Getting Started to Quarkus Messaging with Apache Pulsar]

This guide demonstrates how your Quarkus application can utilize Quarkus Messaging to interact with

[Getting Started to Quarkus Messaging with RabbitMQ]

This guide demonstrates how your Quarkus application can utilize Quarkus Messaging to

Apache Kafka.

Apache Pulsar.

interact with RabbitMQ.

[Getting Started with SmallRye Stork]

The essence of distributed systems resides in the interaction between services.

[Getting Started with gRPC]

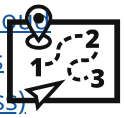
This guide explains how to start using gRPC in your Quarkus application.

[Google Cloud Functions (Serverless)]

This guide explains how you can deploy Quarkus-based Google Cloud Functions.

[Google Cloud Functions (Serverless) with Quarkus REST, Undertow, or Reactive Routes]

This guide explains how you can deploy Vert.x Web, Servlet, or RESTEasy microservices as a Google Cloud Function.



[How dev mode differs from a production application]

How dev mode differs from a production application

[Implementing a gRPC Service]

This guide explains how to implement gRPC services in your Quarkus application.

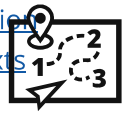
[Kafka Dev UI]

[Initialization tasks]

This reference guide explains how to configure initialization tasks

[Introduction to Contexts and Dependency Injection (CDI)]

Quarkus DI solution is based on the



Dev UI extension for Apache Kafka for development purposes.

[Kubernetes Client]

This guide demonstrates how to use the Fabric8 Kubernetes client to interact with your Kubernetes cluster.

[Kubernetes Config]

Use ConfigMaps as a configuration source for your Quarkus applications.

[Kubernetes extension]

This guide covers how to deploy a native application on Kubernetes.

[Mapping configuration to objects]

Group multiple configuration properties into an object.

[Measuring Performance]

This guide explains how to best measure the footprint of a Quarkus application.

[Measuring the coverage of your tests]

This guide explains how to measure the test coverage of your Quarkus application.

[Migrating to Quarkus REST (formerly RESTEasy Reactive)]

Migrating from RESTEasy Classic to Quarkus REST (formerly RESTEasy Reactive) is straightforward in most cases, however there are a few cases that require some attention.

[Mutiny - Async for mere mortals]

Mutiny is an intuitive, reactive programming library.

[Narayana LRA Participant Support]

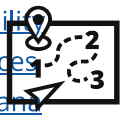
This guides covers the usage of LRA to coordinate activities across services.

[Observability Dev Services]

Entry point for Observability DevServices

[Observability Dev Services with Grafana OTel LGTM]

Instructions on how to use Grafana Otel LGTM



[Packaging And Releasing With JReleaser]

This guide covers packaging and releasing CLI applications using the JReleaser tool.

[Platform]

[Quarkus Base Runtime Image]

[Quarkus Extension]

Learn more about what we call a Platform in the Quarkus world.

To ease the containerization of native executables, Quarkus provides a base image providing the requirements to run these executables.

[Metadata]

Quarkus extensions are distributed as Maven JAR artifacts that application and other libraries may depend on.

[Quarkus Extension Registry]

Learn more about the notion of extension registry and how you can use your own.

[Quarkus Extension for Spring Cache API]

While you are encouraged to use the Cache extension for your application-level caching, Quarkus provides a compatibility layer for Spring Cache in the form of the spring-cache extension.

[Quarkus Extension for Spring DI API]

While you are encouraged to use CDI annotations for injection, Quarkus provides a compatibility layer for Spring dependency injection in the form of the spring-di extension.

[Quarkus Extension for Spring Scheduling API](#)

While you are encouraged to use the Scheduler or Quartz extensions to schedule tasks, Quarkus provides a compatibility layer for Spring Scheduled in the form of the spring-scheduled extension.

[Quarkus Extension for Spring Security API](#)

While you are encouraged to use the Quarkus Security layer to secure your applications, Quarkus provides a compatibility layer for Spring Security in the form of the spring-security extension.

[Quarkus Extension for Spring Web API](#)

While you are encouraged to use Jakarta REST annotations for defining REST endpoints, Quarkus provides a compatibility layer for Spring Web in the form of the spring-web extension.

[Quarkus Messaging Extensions](#)

Event-driven messaging systems have become the backbone of most modern applications, enabling the building of message-driven microservices or complex data streaming pipelines.

[Quarkus Reactive Architecture](#)

Learn more about Quarkus reactive architecture.

[Quarkus Virtual Thread support for gRPC services](#)

This guide explains how to benefit from Java virtual threads when implementing a gRPC service.

[Quarkus Virtual Thread support with Reactive Messaging]

This guide explains how to benefit from Java virtual threads when writing message processing applications in Quarkus.

[Quarkus and Gradle]

Develop and build your Quarkus application with Gradle

[Quarkus and Maven]

Develop and build your Quarkus application with Maven

[Quarkus for the Web]

Learn more about creating all kinds of Web applications with Quarkus.

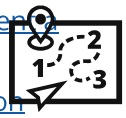
[Qute Templating Engine]

Learn more about how you can use templating in your applications with the Qute template engine.

[RESTEasy Classic]

[Re-augmentation of Quarkus Application](#)

Use mutable jars to rebuild your application with different build time configurations.



[Reactive SQL Clients](#)

This guide covers how to use the Reactive SQL Clients in Quarkus.

[Reading properties from Spring Cloud Config Server](#)

Quarkus provides a compatibility layer for Spring Cloud Config in the form of the spring-cloud-config-client extension.

[Scheduling Periodic Tasks](#)

Modern applications often need to run specific tasks periodically.

[Scheduling Periodic Tasks with Quartz](#)

You need clustering support for your scheduled tasks? This guide explains how to use the Quartz extension for that.

[Scripting with Quarkus](#)

Easy Quarkus-based scripting with jbang.

[Secrets in Configuration](#)

Use encrypted configuration

[Security Testing](#)

This document describes

[Security Tips and Tricks](#)

values to protect sensitive passwords, secrets, tokens and keys.

how to test Quarkus Security.

[Sending emails using SMTP]

Learn more about how you can send email from a Quarkus application with our reactive email client.

[Simplified Hibernate ORM with Panache]

Hibernate ORM is the de facto Jakarta Persistence implementation and offers you the full breadth of an Object Relational Mapper.

[Simplified Hibernate ORM with Panache and Kotlin]

This explains the specifics of using Hibernate ORM with Panache in a Kotlin project.

[Simplified Hibernate Reactive with Panache]

Simplified reactive ORM layer based on Hibernate Reactive.

[Simplified MongoDB with Panache]

This guide covers the usage of MongoDB using active records and repositories.

[Simplified MongoDB with Panache and Kotlin]

This guide covers the usage of MongoDB using active records and repositories in a Kotlin project.

[SmallRye Fault Tolerance](#)

This guide demonstrates how your Quarkus application can utilize the SmallRye Fault Tolerance specification through the SmallRye Fault Tolerance extension.

[SmallRye GraphQL](#)

This guide explains how to leverage SmallRye GraphQL to implement GraphQL services.

[SmallRye GraphQL Client](#)

This guide explains how to leverage SmallRye GraphQL Client to consume GraphQL services.

[SmallRye Health](#)

This guide demonstrates how your Quarkus application can utilize the SmallRye Health extension.

[SmallRye Metrics](#)

This guide demonstrates how your Quarkus application can utilize the SmallRye Metrics extension.

[Testing Your Application](#)

This guide covers testing in JVM mode, native mode, and injection of resources into tests

[Testing components](#)

This reference

[Tips for writing native applications](#)

[Use Hibernate Search in Standalone mode with Elasticsearch/OpenSearch](#)

guide covers the

This guide is a collection of tips to help you solve the problems you encounter when compiling applications to native executable.

Hibernate Search Standalone allows you to index your entities in an Elasticsearch/OpenSearch cluster and easily offer full text search in all your applications, even without Hibernate ORM.

elasticsearch opensearch hibernate search

Use Hibernate Search with Hibernate ORM and Elasticsearch/OpenSearch

Hibernate Search with Hibernate ORM allows you to index your entities in an Elasticsearch/OpenSearch cluster and easily offer full text search in all your Hibernate ORM-based applications.

elasticsearch opensearch hibernate orm search

Using Apache Kafka Streams

This guide demonstrates how your Quarkus application can utilize the Apache Kafka Streams API to implement stream processing applications based on Apache Kafka.

Using Apache Kafka with Schema Registry and Avro

Use Apache Kafka, Avro serialized records, and connect to a schema registry.

Using Apache Kafka with Schema Registry

Using Blaze-Persistence

This guide explains how to use Blaze-

Using Eclipse Vert.x API from a Quarkus Application

and JSON Schema

Use Apache Kafka, Json Schema serialized records, and connect to a schema registry.

Persistence to simplify your data and DTO layers.

This guide explains how to use Vert.x in Quarkus to build reactive applications.

vertx event verticle

Using Flyway

This guide covers how to use the Flyway extension to manage your schema migrations.

Using Hibernate ORM and Jakarta Persistence

Hibernate ORM is the de facto Jakarta Persistence implementation and offers you the full breath of an Object Relational Mapper.

Using Hibernate Reactive

Hibernate Reactive is a reactive API for Hibernate ORM, supporting non-blocking database drivers and a reactive style of interaction with the database.

Using JMS

This guide demonstrates how your

Using JWT RBAC

This guide explains

Using Java Flight Recorder

Quarkus application can use JMS messaging with AMQP 1.0 using Apache Qpid JMS, or using Apache ActiveMQ Artemis JMS.

how your application can use SmallRye JWT to provide secured access to the Jakarta REST endpoints.

This guide explains how Java Flight Recorder (JFR) can be extended to provide additional insight into your Quarkus application.

Using Keycloak Admin Client

The Quarkus Keycloak Admin Client and its reactive twin support Keycloak Admin Client which can be used to configure a running Keycloak server.

sso oidc security keycloak

Using Kotlin

This guide explains how to use Kotlin.

Using Liquibase

This guide covers how to use the Liquibase extension to manage your schema migrations.

[Using Liquibase MongoDB]

Liquibase is an open source tool for database schema change management, it allows managing MongoDB databases via it's MongoDB Extension.

[Using OAuth2 RBAC]

This guide explains how your Quarkus application can utilize OAuth2 tokens to provide secured access to the Jakarta REST endpoints.

oauth

[Using OpenAPI and Swagger UI]

This guide explains how to use the OpenAPI extension to generate an OpenAPI descriptor and get a Swagger UI frontend to test your REST endpoints.

[Using OpenTelemetry Logging]

This guide explains how your Quarkus application can utilize OpenTelemetry Logging to provide centralised logging for interactive web applications.

[Using OpenTelemetry Metrics]

This guide explains how your Quarkus application can utilize OpenTelemetry to provide metrics for interactive web applications.

[Using OpenTelemetry Tracing]

This guide explains how your Quarkus application can utilize OpenTelemetry Tracing to provide distributed tracing for interactive web applications.

[Using Podman with Quarkus]

Podman is an open-source, daemonless, and rootless container engine for developing, managing, and running OCI Containers on Linux, Windows and Mac.

[Using Reactive Routes]

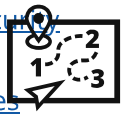
This guide demonstrates how to use reactive routes.

[Using SSL With Native Executables]

In this guide, we will discuss how you can get your native images to support SSL, as native images don't support it out of the box.

[Using Security with .properties File]

Quarkus provides support for properties file-based authentication intended for development and testing purposes.



[Using Security with JDBC]

This guide demonstrates how your Quarkus application can use a database to store your user identities.

[Using Security with WebAuthn]

This guide demonstrates how your Quarkus application can use WebAuthn authentication instead of passwords.

[Using Security with an LDAP Realm]

[Using Software Transactional Memory in Quarkus]

[Using Stork with Kubernetes]

This guide explains

This guide demonstrates how your Quarkus application can use a LDAP directory to store your user identities.

This guides covers the usage of Software Transactional Memory (STM).

how to use Stork with Kubernetes for service discovery and load balancing.

Using WebSockets with Undertow

This guide explains how your Quarkus application can utilize web sockets to create interactive web applications.

Using a Credentials Provider

This guides explains how to use the Vault credentials provider or implement your own custom one.

Using gRPC CLI

This page explains how to use gRPC CLI.

Using the Cassandra Client

This guide covers how to use the Apache Cassandra NoSQL database in Quarkus.

Using the Infinispan Client

This guide covers how to use Infinispan with Quarkus.

Using the MongoDB Client

This guide covers how to use MongoDB in Quarkus.

[Using the REST Client]

This guide explains how to use the REST Client.

[Using the Redis Client]

This guide covers how to use a Redis datastore in Quarkus.

[Using the event bus]

This guide explains how different beans can interact using the event bus.

vertx
vert.x

[Using the legacy REST Client]

This guide explains how to use the RESTEasy Classic REST Client in order to interact with REST APIs (JSON

[Using the legacy REST Client with Multipart]

This guide explains how to use the RESTEasy Classic REST Client to send multipart REST requests, typically to upload documents.

[Using xDS gRPC]

This page explains how to enable xDS gRPC usage in your Quarkus application.

and other) with very little effort.

[Validation with Hibernate Validator]

This guide covers how to use Hibernate Validator/Bean Validation in your REST services.

[Web dependency locator]

Learn more about how to use the web dependency locator

[Writing JSON REST Services]

JSON is now the lingua franca between microservices.

[Writing REST Services with Quarkus REST (formerly RESTEasy Reactive)]

Discover how to develop highly scalable reactive REST services with Jakarta REST and Quarkus REST.

[Writing Your Own Extension]

Quarkus extensions optimize your applications by pushing as much work as possible to the build operation.

[gRPC]

Entry point for everything gRPC.

Navigation

Home
About
Blog
Podcast
Events
Newsletter
User Stories
Roadmap
Security policy
Usage
Brand
Wallpapers
Privacy Policy

Follow Us

X
Bluesky
Mastodon
Threads
Facebook
Linkedin
Youtube
GitHub

Get Help

Support
Guides
FAQ
Get Started
Stack Overflow Discussions
Development mailing list

Languages

English
Português (Brasileiro)
Español
简体中文
日本語

Quarkus is made of community projects

Eclipse Vert.x
SmallRye
Hibernate
Netty
RESTEasy
Apache Camel
Eclipse MicroProfile
And many more...

This website was built with Jekyll, is hosted on GitHub Pages and is completely open

source. If you want to make it better, fork the website and show us what you've got.

Sponsored by