# Quarkus

Quarkus is a Kubernetes-native Java framework designed for building cloud-native, container-first applications. It is optimized for both GraalVM and HotSpot, enabling native image compilation as well as traditional JVM deployment. With a strong focus on fast startup times and low memory consumption, Quarkus is particularly suitable for microservices and serverless environments. The framework supports live coding, giving developers immediate feedback without restarting applications, and follows a zero-configuration philosophy to reduce boilerplate. Quarkus offers robust dependency injection support via CDI and integrates seamlessly with industry-standard Java APIs and libraries such as JAX-RS, Hibernate, JPA, RESTEasy, Jackson, and Kafka. Reactive programming is natively supported using Mutiny and Vert.x. When compiled to native images via GraalVM, Quarkus applications benefit from drastically improved cold-start performance and minimal resource usage. Applications can be packaged as JARs or as native executables depending on deployment needs. Quarkus also provides Dev Services, which automatically launch required dependencies during development. Configuration is centralized and flexible through application.properties or YAML files. Development is further simplified by a CLI and Maven/Gradle plugins for scaffolding and lifecycle management. It includes built-in support for OpenAPI, Swagger UI, and REST clients to streamline API-driven development. For persistence, Quarkus introduces Panache, simplifying data access with a more expressive JPA layer. Comprehensive security support covers JWT, OAuth2, and role-based access control. The framework is highly suited for deployment on platforms like Kubernetes, OpenShift, and Knative. A modular extension system allows developers to include only the features they need. Backed by an active and growing community, Quarkus continues to evolve rapidly and enjoys broad adoption, especially in microservice architectures.