Spring Data Couchbase Datasheet

Key benefits

- Spring configuration support using Java-based @Configuration classes
- CouchbaseTemplate helper class for performing common Couchbase operations
- Annotation-based metadata mapping
- Automatic implementation of repository interfaces (backed by Couchbase's SQL-based query language, N1QL)
- Can be used as the backend for @Cacheable support, to cache any objects needed for high-performance access
- Query derivations
- JMX administration and monitoring

Why Couchbase

Couchbase is a modern database for enterprise applications. It delivers unmatched versatility, performance, scalability, and financial value across cloud, on-premises, hybrid, distributed cloud, and edge computing deployments.

Couchbase offers support for the flexible topologies needed to develop and maintain business-critical applications in any environment (on-prem, cloud, hybrid). For example, Couchbase scopes give developers the power to create and modify schemas on the fly to enable multi-tenancy use cases along with full transaction support via code or via SQL syntax.

Why use spring data with Couchbase

During application development, implementing the data layer can require complex and often repetitive boilerplate code. Spring Data handles most of the database access complexity, and object-relational mappings drastically reduce the amount of boilerplate code required. This platform (see figure 1) improves the maintainability, scalability, and performance of the persistence layer.

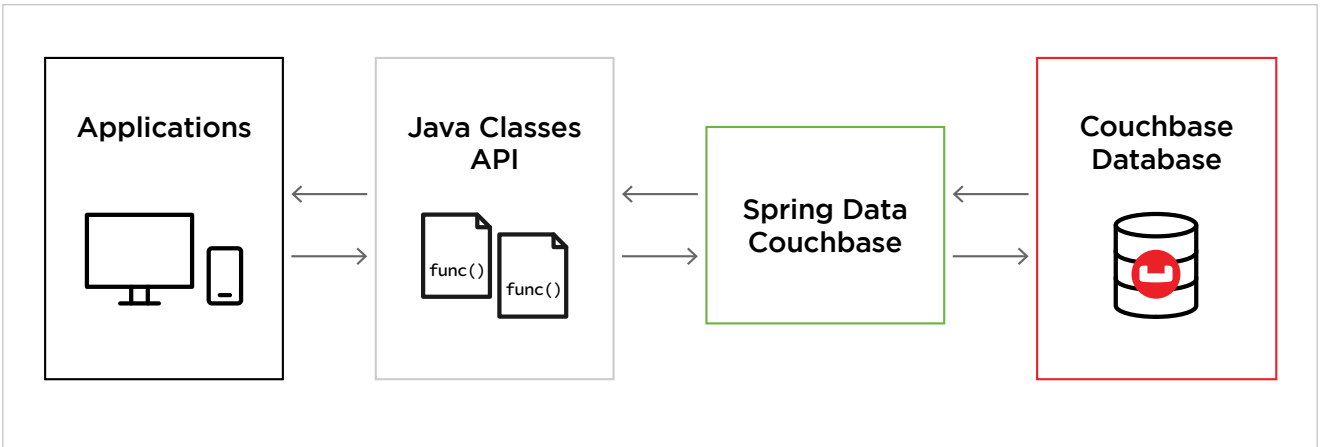

Figure 1. Spring Data Couchbase model

The Couchbase Spring Data project makes it easy to use Couchbase within a modern data access framework. Many Java programmers are familiar with the Spring framework, so utilizing it with Couchbase is a natural progression. Spring Data Couchbase provides seamless integration with the Couchbase database.

Key functional areas of Spring Data Couchbase are a POJO-centric model for interacting with Couchbase buckets and easily writing a repository-style data access layer. In addition, features like query derivations enhance developer productivity and reduce the learning curve by abstracting the underlying logic.

Summary

Utilizing the Spring Data Couchbase project is straightforward. Java programmers can code with all of the tools from the Spring ecosystem while leveraging the speed and flexibility of Couchbase. This makes it easier to build Spring-powered applications that use new data access technologies such as non-relational databases like Couchbase NoSQL and cloud-based data services like Couchbase Cloud™.

Prerequisites and getting started

Prerequisite: Run Couchbase Server

- Install and configure Couchbase Server
  - A bucket is created
  - A primary index is created and built on the bucket
- Install and configure developer's preferred IDE (integrated developer environment)
- Install Java 8 or higher

Getting Started

- Create a Spring Data project from your preferred IDEVV or bootstrap your application with Spring Initializr

Take the next step

To learn more please visit the Couchbase website.

Spring Data Couchbase

Spring Data Couchbase Reference Documentation

See the section on using a snapshot/milestone build. The latest features with transactions and scopes are in 4.3.0-M2 (Pascal) and later.

Couchbase Server Quickstart – Java with Spring Data Couchbase and IntelliJ

Quickstart in Couchbase with Java and Spring Boot

Tutorial: Boosting Spring Data Performance with Couchbase

At Couchbase, we believe data is at the heart of the enterprise. We empower developers and architects to build, deploy, and run their mission-critical applications. Couchbase delivers a high-performance, flexible and scalable modern database that runs across the data center and any cloud. Many of the world's largest enterprises rely on Couchbase to power the core applications their businesses depend on.

For more information, visit www.couchbase.com.