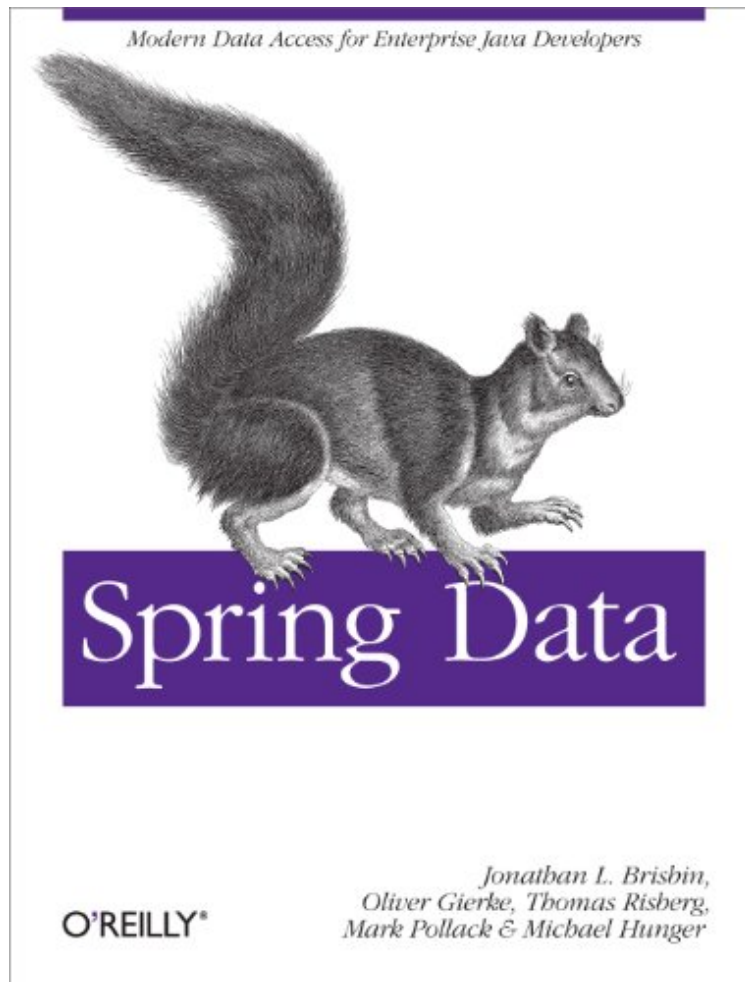


Spring Data: Modern Data Access for Enterprise Java

Von Mark Pollack, Oliver Gierke, Thomas Risberg, Jon Brisbin, Michael Hunger
DOC | *audiobook | ebooks | Download PDF | ePub



Produktinformation -Verkaufsrank: #363472 in eBooksVerffentlicht am: 2012-10-12Erscheinungsdatum:
2012-10-12File Name: B009PSWJHA | File size: 54.Mb

Von Mark Pollack, Oliver Gierke, Thomas Risberg, Jon Brisbin, Michael Hunger : Spring Data: Modern Data Access for Enterprise Java before purchasing it in order to gage whether or not it would be worth my time, and all praised Spring Data: Modern Data Access for Enterprise Java:

KundenrezensionenHilfreichste Kundenrezensionen1 von 1 Kunden fanden die folgende Rezension hilfreich. Great introduction into Spring Data and associated frameworksVon Michael PloedThe book is written by the core developers of the Spring Data Framework. In my eyes it served me very well getting started with Spring Data as well as in diving into more advanced topics such as the integration of QueryDSL for different technologies such as JPA or plain JDBC. In addition to that nearly two thirds of the book cover the support for NoSQL and Big Data Stores such as MongoDB, Neo4j, Redis or Hadoop.Two smaller aspects are to criticize about the book: I would have liked to have an example for a Gradle based integration of QueryDSL and the chapter about Rapid Application Development is

between the NoSQL and the Big Data chapters. It should have been put at the end of the book instead. Overall you get a great introduction into the framework and some other new frameworks surrounding it.

0 von 0 Kunden fanden die folgende Rezension hilfreich. Data access the Spring way Von Alex

The book provides a comprehensive overview of the Spring Data project with lots of helpful examples. Starting with a brief chapter on JDBC and relational databases, the reader is quickly introduced to the world of NoSQL data stores, including MongoDB, Neo4J and Redis. Focal to Spring's philosophy of data access are templates and repositories. While templates are used to manage the underlying resources and to map vendor-specific exceptions into Spring's own `DataAccessException` hierarchy, repositories are used to provide CRUD functionality and more sophisticated queries for data access. The introduced data stores structure data in different ways, either as a graph, as keys and values or document-based. Since each store has its own specifics, the framework does not try to hide access to different stores behind a single API (similar to JPA), but rather provides a consistent programming model to rely upon. Building upon the repository abstraction, the framework is also able to infer queries from method names following certain conventions, freeing the client from writing code, which can be largely considered boilerplate. In addition, Spring Data integrates nicely with Querydsl enabling the use of type safe queries, thereby reducing the risk of typos sneaking into your queries.

Having introduced the mere data access, the remainder of the book deals with several applications of the capabilities provided by Spring Data, such as the integration into Spring's rapid application development tool Roo, or the facility of exporting repositories via a RESTful API which comes in handy for many data-driven backends. A topic which has gained more and more attention recently, and which is also tightly related to NoSQL is Big Data. Several examples show, how large volumes of data can be processed and stored using Hadoop and adjacent Spring modules, including Spring Integration and Spring Batch, before the book concludes with a brief introduction to GemFire, VMware's platform for distributed data management. Although prior knowledge of the framework is beneficial in order to put Spring Data in larger context and appreciate how the framework remains true to its philosophy of being non-intrusive, the examples make it easy even for beginners to follow the narrative.

Kurzbeschreibung You can choose several data access frameworks when building Java enterprise applications that work with relational databases. But what about big data? This hands-on introduction shows you how Spring Data makes it relatively easy to build applications across a wide range of new data access technologies such as NoSQL and Hadoop. Through several sample projects, you'll learn how Spring Data provides a consistent programming model that retains NoSQL-specific features and capabilities, and helps you develop Hadoop applications across a wide range of use-cases such as data analysis, event stream processing, and workflow. You'll also discover the features Spring Data adds to Springs existing JPA and JDBC support for writing RDBMS-based data access layers.

Learn about Springs template helper classes to simplify the use of database-specific functionality
Explore Spring Data's repository abstraction and advanced query functionality
Use Spring Data with Redis (key/value store), HBase (column-family), MongoDB (document database), and Neo4j (graph database)
Discover the GemFire distributed data grid solution
Export Spring Data JPA-managed entities to the Web as RESTful web services
Simplify the development of HBase applications, using a lightweight object-mapping framework
Build example big-data pipelines with Spring Batch and Spring Integration