1.	Record Nr.	UNINA9910299050103321
	Autore	Soukup Jiri
	Titolo	Serialization and Persistent Objects : Turning Data Structures into Efficient Databases / / by Jiri Soukup, Petr Macháek
	Pubbl/distr/stampa	Berlin, Heidelberg : , : Springer Berlin Heidelberg : , : Imprint : Springer, , 2014
	ISBN	3-642-39323-3
	Edizione	[1st ed. 2014.]
	Descrizione fisica	1 online resource (276 p.)
	Disciplina	004 004.5 005.43 005.74
	Soggetti	Data structures (Computer science) Database management Operating systems (Computers) Data Storage Representation Database Management Operating Systems
	Lingua di pubblicazione	Inglese
	Formato	Materiale a stampa
	Livello bibliografico	Monografia
	Note generali	Description based upon print version of record.
	Nota di contenuto	Introduction -- Fundamentals of persistence -- Data structures, libraries, and UML -- Advanced features, schema migration -- Languages, their features and limitations -- Automatic persistence for Objective-C -- Benchmark -- Proposal to add a keyword to all OO languages -- The future. .
	Sommario/riassunto	Recently, the pressure for fast processing and efficient storage of large data with complex relations increased beyond the capability of traditional databases. Typical examples include iPhone applications, computer aided design – both electrical and mechanical, biochemistry applications, and incremental compilers. Serialization, which is sometimes used in such situations is notoriously tedious and error prone. In this book, Jiri Soukup and Petr Macháek show in detail how to write programs which store their internal data automatically and transparently to disk. Together with special data structure libraries

which treat relations among objects as first-class entities, and with a UML class-diagram generator, the core application code is much simplified. The benchmark chapter shows a typical example where persistent data is faster by the order of magnitude than with a traditional database, in both traversing and accessing the data. The authors explore and exploit advanced features of object-oriented languages in a depth hardly seen in print before. Yet, you as a reader need only a basic knowledge of C++, Java, C#, or Objective C. These languages are quite similar with respect to persistency, and the authors explain their differences where necessary. The book targets professional programmers working on any industry applications, it teaches you how to design your own persistent data or how to use the existing packages efficiently. Researchers in areas like language design, compiler construction, performance evaluation, and no-SQL applications will find a wealth of novel ideas and valuable implementation tips. Under http://www.codefarms.com/book, you will find a blog and other information, including a downloadable zip file with the sources of all the listings that are longer than just a few lines – ready to compile and run. .