

1. Record Nr.	UNINA9911020025403321
Autore	Cherubini Umberto
Titolo	Structured finance : the object oriented approach // Umberto Cherubini, Giovanni Della Lunga
Pubbl/distr/stampa	Chichester, England ; ; Hoboken, NJ, : John Wiley & Sons, c2007
ISBN	9786610856046 9781119209263 1119209269 9781280856044 1280856041 9780470512722 0470512725
Edizione	[1st ed.]
Descrizione fisica	1 online resource (300 p.)
Collana	Wiley finance series
Altri autori (Persone)	Della LungaGiovanni
Disciplina	332.63/27
Soggetti	Structured notes (Securities) Derivative securities Investment analysis - Mathematical models Financial engineering
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Structured Finance; Contents; 1 Structured Finance: A Primer; 1.1 Introduction; 1.2 Arbitrage-free valuation and replicating portfolios; 1.3 Replicating portfolios for derivatives; 1.3.1 Linear derivatives; 1.3.2 Nonlinear derivatives; 1.4 No-arbitrage and pricing; 1.4.1 Univariate claims; 1.4.2 Multivariate claims; 1.5 The structuring process; 1.5.1 The basic objects; 1.5.2 Risk factors, moments and dimensions; 1.5.3 Risk management; 1.6 A tale of two bonds; 1.6.1 Contingent coupons and repayment plans; 1.6.2 Exposure to the risky asset; 1.6.3 Exposure to volatility; 1.6.4 Hedging 1.7 Structured finance and object-oriented programmingReferences and further reading; 2 Object-Oriented Programming; 2.1 Introduction; 2.2 What is OOP (object-oriented programming)?; 2.3 Analysis and design; 2.3.1 A simple example; 2.4 Modelling; 2.4.1 The Unified Modelling Language (UML); 2.4.2 An object-oriented programming

language: Java; 2.5 Main ideas about OOP; 2.5.1 Abstraction; 2.5.2 Classes; 2.5.3 Attributes and operations: the Encapsulation principle; 2.5.4 Responsibilities; 2.5.5 Inheritance; 2.5.6 Abstract classes; 2.5.7 Associations; 2.5.8 Message exchanging; 2.5.9 Collections
2.5.10 PolymorphismReferences and further reading; 3 Volatility and Correlation; 3.1 Introduction; 3.2 Volatility and correlation: models and measures; 3.2.1 Implied information; 3.2.2 Parametric models; 3.2.3 Realized (cross)moments; 3.3 Implied probability; 3.4 Volatility measures; 3.4.1 Implied volatility; 3.4.2 Parametric volatility models; 3.4.3 Realized volatility; 3.5 Implied correlation; 3.5.1 Forex markets implied correlation; 3.5.2 Equity "average" implied correlation; 3.5.3 Credit implied correlation; 3.6 Historical correlation; 3.6.1 Multivariate GARCH
3.6.2 Dynamic correlation model3.7 Copula functions; 3.7.1 Copula functions: the basics; 3.7.2 Copula functions: examples; 3.7.3 Copulas and survival copulas; 3.7.4 Copula dualities; 3.8 Conditional probabilities; 3.9 Non-parametric measures; 3.10 Tail dependence; 3.11 Correlation asymmetry; 3.11.1 Correlation asymmetry: finance; 3.11.2 Correlation asymmetry: econometrics; 3.12 Non-exchangeable copulas; 3.13 Estimation issues; 3.14 Levy processes; References and further reading; 4 Cash Flow Design; 4.1 Introduction; 4.2 Types of bonds; 4.2.1 Floaters and reverse floaters
4.2.2 Convertible bonds4.2.3 Equity-linked notes; 4.2.4 Inflation-linked bonds; 4.2.5 Asset-backed securities; 4.3 Time and scheduler issues; 4.3.1 Payment date conventions; 4.3.2 Day count conventions and accrual factors; 4.4 JScheduler; 4.4.1 Date handling in Java; 4.4.2 Data models; 4.4.3 Design patterns; 4.4.4 The Factory Method pattern; 4.5 Cash flow generator design; 4.5.1 UML's activity diagram; 4.5.2 An important guideline to the data model for derivatives: FpML; 4.5.3 UML's sequence diagram; 4.6 The cleg class; References and further reading; 5 Convertible Bonds; 5.1 Introduction
5.2 Object-oriented structuring process

Sommario/riassunto

Structured Finance: The Object Orientated Approach is aimed at both the finance and IT professionals involved in the structured finance business with the intention of sharing common concepts and language within the industry. The financial community (structurers, pricers and risk managers) view structured products as collections of objects under the so-called replicating portfolio paradigm. The IT community use object oriented programming (OOP) techniques to improve the software updating and maintenance process. For them structured products are collections of objects as well. Despite use
