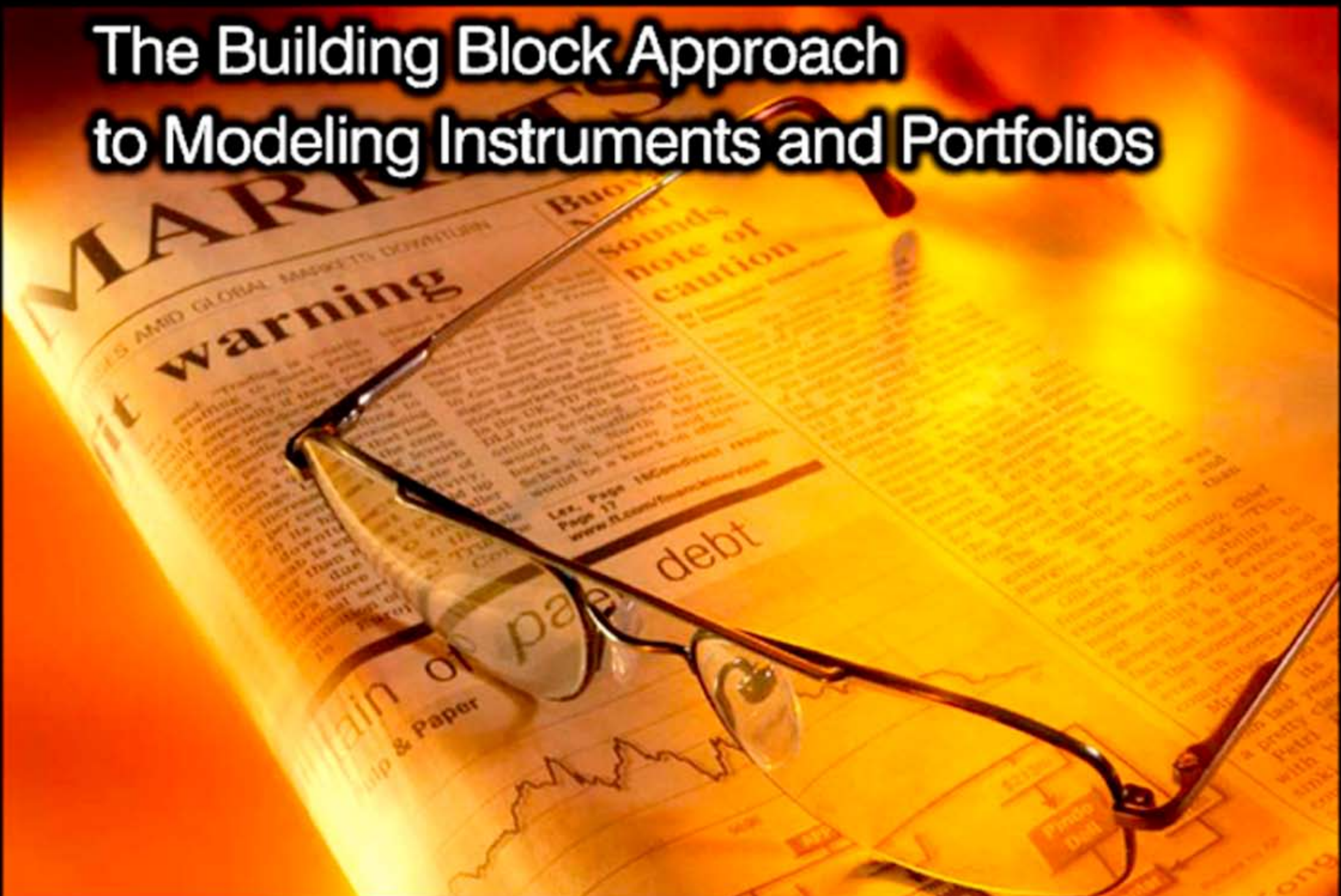
THE ELSEVIER AND MONDO VISIONE WORLD CAPITAL MARKETS SERIES

Pricing, Risk, and Performance Measurement in Practice

The Building Block Approach to Modeling Instruments and Portfolios



Wolfgang Schwerdt
Marcelle von Wendland

ELSEVIER

Advance Praise for Pricing, Risk, and Performance Measurement in Practice

"The book represents a fresh and innovative departure from 'traditional' approaches to modelling of securities data. Subsequently, it also presents much more flexible ways to analyze and process the data. Even if you are not involved with re-architecting an organization's master data handling, there are numerous ideas, principles, and nuggets that make it a worthwhile read."

–Dr. Matthias Autrata, Director, Deutsche Asset Management Global Technology Operations—Asset Management Platform Services

"The piece of work is a very nice, well-structured step-to-step guide on how to valuate securities. The authors have achieved in making a highly complex topic easy to understand, and offer a guide for valuating securities from the very beginning to the very end."

–Alexandra Balloff, Executive Director, Global Head of Market Data Management, WestLB

"It is refreshing to see an approach to financial data modeling which adequately reflects the semantics of the instruments themselves. This book presents complex mathematical models for securities calculations, illustrated with easy to understand diagrams. This is combined with a mature and comprehensive method for developing systems and data models that should be followed by any financial institution that wishes to develop robust and maintainable systems for risk, valuation, and performance attribution."

–Mike Bennett, Hypercube Limited, EDM Council Semantics Repository Architect

"By taking from the outset a technical and applied perspective, this book greatly complements the theoretical literature. Practitioners will obviously like it, but also all others interested in valuation and risk control should find the book stimulating as providing insights on the real issues one faces when implementing models."

–Ulrich Bindseil, Head of Risk Management Division, European Central Bank

"I found the approach of the authors to be really interesting. The book combined with the available web support will provide anyone interested in the subject with a compendium of useful information and ideas that they will find invaluable."

–Dennis Cox, CEO, Risk Reward Limited; Chairman, Risk Forum Securities and Investment Institute

"*Pricing, Risk, and Performance Measurement in Practice* is very ambitious in scope yet succeeds by moving step-by-step from the fundamentals of modelling to a comprehensive overview of valuation, risk, and performance management issues. By anchoring the discussion firmly in methodology and by mirroring theory chapters with sections on practical aspects of implementation, the authors offer a valuable 'real life' toolbox to anyone working in the financial services industry."

–Martijn Groot, Director, Market Strategy, Asset Control, Luxembourg

"I endorse the publication of Wolfgang Schwerdt and Marcelle von Wendland's *Pricing, Risk, and Performance Measurement in Practice*. This book gives interesting and valuable insights for the practitioner to model instruments and portfolios. Following the building block approach, the authors demonstrate in a convincing manner that complex situations can be decomposed in an easy and flexible way. Concrete figures and a lot of examples help to explain the basic ideas and how to handle problems in practical situations."

–Olaf Huebler, Professor of Econometrics in the Institute of Empirical Economics at the Leibniz University Hannover and Research Fellow of IZA (Bonn) and IAB (Nuremberg)

"The authors show that risk management needs to be based on the economic properties of financial instruments and that a vital link exists between data, data modeling, and risk management. To my knowledge this is the first time this link is explicitly demonstrated in the literature on valuation and risk management. Overall an interesting and demanding book for everybody who intends to build or improve risk management systems."

–Martin Janssen, Banking Institute, University of Zurich; CEO ECOFIN Research and Consulting Ltd., Zurich, Switzerland

"The authors have achieved a perfect balance between theory and the practical implementation of financial information management and modeling tools. For anyone with a need to create or improve performance and risk management systems, this book offers a road map to get you there quickly and correctly."

–Richard M. King, Managing Director, LeftBrain, Inc.

"The authors of *Pricing, Risk and Performance Measurement* are skillfully bridging the gap between risk management theory and its practical real life application. Whether you are developing and applying ad-hoc risk models or finding yourself faced with building an enterprise-wide performance analytics platform, their book will provide you with a robust and proven toolkit to do so.

Readers are equipped with a thoroughly researched best practice guide and will profit from step-by-step guidance through all stages of planning, composing, mapping, creating, calibrating, and refining a solid risk portfolio model. Whilst the vast number of tables, graphical illustrations, process diagrams, and sample calculations provide a great stand-alone desktop reference, the combination with the book's software tools and templates on its companion website make this an invaluable aid.

The book is unique as it introduces and advocates a proven methodology which was developed together with and has been successfully applied by many banks to model over five million financial instruments—what better 'seal of approval' can you ask for?"

–Markus Krebsz, Capital Markets Consultant, SME Panel member for Credit Risk & Structured Products, 4Most

"This book is the first to point out the fundamental link between data, data modeling, and risk management. It contains a huge number of very valuable details and concepts. A must read for anyone setting out to build risk management systems."

–Didier Maman, Managing Director, Vidatis s.a.r.l.

"In times of financial turmoil survival depends on knowing where your risks are and having the right tools to lead you to where you want to be going. This great book is just about that—a must read."

–Mark Michel, Wincor Nixdorf Banking Solutions

"Are you always thinking that everything except plain vanilla is too complicated? Try this book and you realize that even the most complex systems are built piece by piece from simple building blocks. This book is a magnificent tool for financial instrument expert and IT-expert to understand each other. *Pricing, Risk, and Performance Measurement* creates value-added to design new systems from practical and user-friendlier perspective."

–Marko Myller, Economist, Oversight of Market Infrastructure, Financial Markets and Statistics, Bank of Finland

"The book reveals the authors' profound exposure to securities database projects and distinctively considers risk concepts in an integrated manner. The presented building block approach supports flexibility in data models—an important prerequisite in times of change. A must read for all those who would want to cross the border from pure theory to practice, and like to know more about the more practical aspects of statistics compilation, thus including the latest state-of-knowledge in international securities database projects."

–Robert Obrzut, Economist-Statistician, Banque centrale du Luxembourg

"The book by Wolfgang Schwerdt and Marcelle von Wendland describes very clearly the Building Block approach to financial instrument portfolio modelling. This innovative method provides for flexibility in modelling of large portfolios containing instruments of different characteristics, and is open, unlike standard database approach, to adoption of new features or parameters that may be necessary in the future. The authors are among the most experienced persons in the field of theory and practical use of Building Block approach, their book shows a new direction and is full of practical solutions for all persons dealing with portfolio and risk management. A valuable position."

–Marcin Sienicki, Portfolio Investment Statistics Division, Statistics Department, National Bank of Poland

"Recent events have shown how financial information is key. This book is a guide to manage it. Useful both for new traders entering the business and for big organisations, who want to check if their system is up-to-date."

–Fabio Salvatore Piamonte, Reference data on entities and financial instrument division, Banca d'Italia

"This is the book the data management community has been waiting for! The Building Block approach introduced by Wolfgang and Marcelle de-mystifies the process of market risk modelling; they rightly consider the operational and business risk associated with the more complex financial instruments and the need to source a broader spectrum of high quality reference data to meet the needs of the entire organisation. I would recommend this highly."

–Lisa Sully, Global Head of Data Management, Aberdeen Asset Management PLC

"This book is the most comprehensive I have come about in terms of linking business and business requirements to IT implementation. And on top you get ready-to-use components and advice—should be a must read for anyone embarking on risk management."

–Rainer Zahradnik, Head of Software Development at RTC, Switzerland

The Elsevier and Mondo Visione World Capital Markets Series

ELSEVIER

mondovisione
Worldwide Exchange Intelligence

Series Editor: Herbie Skeete

The Elsevier and Mondo Visione World Capital Markets Series consists of books that cover the developments in the capital markets, as well as basic texts introducing the markets to those working directly or indirectly in the capital markets. Investors are more knowledgeable and demanding than ever before and there is a thirst for information by professional investors and those who sell and provide services to them. Regulators are demanding more transparency and new rules and regulations are being introduced constantly. The impact of competition means that markets are constantly changing and merging, and new instruments are being devised. The Series provides cutting-edge information and discussion about these and other developments affecting the capital markets. Technology underpins and is driving innovation in the markets. Inappropriate technology or no technology can bring down even the soundest financial institution. This series therefore also includes books that enable market experts to understand aspects of technology that are driving the markets.
Books in the Series include:

- *The Future of the Financial Exchanges* by Herbie Skeete
- *Managing Financial Information in the Trade Lifecycle* by Martijn Groot
- *MDDL and the Quest for a Market Data Standard* by Martin Sexton
- *Market Data Explained* by Marc Alvarez
- *Pricing, Risk, and Performance Measurement in Practice* by Wolfgang Schwerdt and Marcelle von Wendland

Series Editor Herbie Skeete is a well-known figure in the financial information industry having spent twenty-six years at Reuters. During his many senior positions with Reuters—most recently as Head of Equities Content and Head of Exchange Strategy—Mr. Skeete has become recognized globally as an expert on exchanges and content issues. He is frequently asked to address conferences and to contribute to roundtable discussions. Mr. Skeete runs the exchange information publisher Mondo Visione Ltd; edits the industry-standard *Handbook of World Stock, Commodity, and Derivatives Exchanges*, which celebrates its seventeenth edition in 2007; and operates the exchange information website www.exchange-handbook.com.

Pricing, Risk, and Performance Measurement in Practice

The Building Block Approach to Modeling Instruments and Portfolios

Wolfgang Schwerdt
Marcelle von Wendland

ELSEVIER

AMSTERDAM · BOSTON · HEIDELBERG · LONDON
NEW YORK · OXFORD · PARIS · SAN DIEGO
SAN FRANCISCO · SINGAPORE · SYDNEY · TOKYO

mondovisione
Worldwide Exchange Intelligence

Elsevier
84 Theobald's Road, London WC1X 8RR, UK
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA
525 B Street, Suite 1900, San Diego, California 92101-4495, USA

Library of Congress Cataloging-in-Publication Data
Application submitted

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

ISBN: 978-0-12-374521-7

For information on all Academic Press publications
visit our website at www.elsevierdirect.com

Printed in the United States of America
09 10 11 12 13 7 6 5 4 3 2 1

Contents

About the Authors

Holding a PhD in applied Econometrics, **Wolfgang Schwerdt** worked for seven years for the European Central Bank on the Centralised Securities Database (CSDB) project. Today, Wolfgang is based in Geneva, Switzerland, working on a start-up project in internet finance. He can be contacted via email at wolfgang.schwerdt@googlemail.com

Marcelle von Wendland, MSI has over 15 year of experience in the investment management and banking industry, specializing in portfolio accounting, performance, and risk management. She studied Economics and International Relations at the London School of Economics. Marcelle holds a number of professional qualifications in the area of Risk and Investment Management and is a very active member of the Chartered Institute for Securities & Investment where she was, until recently, deputy chairwoman of the Risk Forum. Marcelle is a co-founder and director of Bancstreet Capital Partners Ltd, and Vice President at Fincore Ltd, a fast-growing, medium-sized software house headquartered in London, UK. She can be reached via email at mvw@bancstreet.com

Preface

If you are reading this you are probably either considering buying this book or have already done so.

If you have already purchased this book we would like to say thank you, and congratulate you for deciding to join us on our journey toward better, faster, more resilient, and more flexible pricing, risk modelling, and performance measurement solutions.

If you are also fairly new to or just getting acquainted with pricing, risk modelling, and performance measurement we hope that this book will help you hit the ground running when you implement your first models. You will benefit regardless of whether your first model implementations are part of your studies or you have been dropped in at the deep and your models are for real-life decision making at an investment management firm or bank.

If you have not yet purchased this book you are probably grappling with the tough choice of picking the right book for yourself from the many great books on modelling financial instruments and portfolios that fill the shelves in front of you at the bookshop. Now let's make your choice a bit easier: If you are looking for a book on the theoretical foundations of modern finance or something similar you can now safely put this book away and try your luck by reaching for another book on the shelf. If, however, you need or want to build practical pricing, risk, or performance measurement models that stand up to the harsh demands of real life in the investment, risk management, or banking world, then hold on tight to the book in your hands!

This book is intensely practical and has come from the authors' many years of experience with financial instrument and portfolio models. That experience spans from small ad hoc research applications to very large enterprise-level models that are able to deal with hundreds of thousands or even millions of instruments and investment positions, tight processing schedules, often less than perfect input data, and high demands on accuracy and robustness.

The building block method explained in this book was developed for the financial software solutions vendor Fincore Financial. It is the culmination of many years of practical research in the field of financial information technology. The approach is supported by leading-edge commercial software such as Fincore Financial's Investment Information Central and provides unique support for the needs of buy side, hedge fund, and risk management requirements.

The building block approach is also at the very heart of Fincore Financial's commercial off-the-shelf solutions for investment management and banking. The data models and model mappings in this book have been made available by Fincore Financial. Although we have simplified them and renamed some of the attributes and classes to better highlight the key aspects of interest in this book, they are a true subset of Fincore Financial's successful commercial software products.

Numerous worked examples in this book and an extensive collection of downloadable example programs and ready-to-run source code makes this book the ideal companion for anyone who needs to implement pricing, risk, or performance measurement software solutions.

The book provides you with many practical examples of how to represent financial instruments, portfolios, valuation, risk- and performance model inputs and outputs, always using the building-block approach-based data model set out in the book. It shows many practical examples of how to wire up common, valuation, risk- and performance models with data in the data mode.

Fincore Financial is also sponsoring the companion web site for this book, which is another very valuable practical tool that is part of this book. It contains a wealth of practical examples and updated information, and will help you get the most out of this book. The companion web site provides a rich source of tools, test and sample data, tips, tricks, and further updated information.

How to Use the Companion Web Site

The companion web site is an essential and valuable part of this book and you should access and use it to get the most from reading. The companion web site is exclusive for the readers of this book and entirely free to use.

The companion web site can be found at http://modelbook.fincorefinancial.com/.
It contains:

- The data, code, and tutorial notes for all Lab Exercises mentioned in this book
- A lot of ready-to-use example code in ORACLE PL/SQL, R and AMPL
- Up-to-date information on where to get the free companion tools for running the code examples: R and AMPL modelling environments and Personal ORACLE Database

- Other useful and updated information on modelling and model implementation
- Information on contacting the authors directly if you have any further questions or comments
- Other useful information and additional useful tools

Free Companion Software: R Language

Many of the examples in this book are written in **R**, which is an open source modelling software solution and language for statistics and mathematical modelling similar to the well known S-Plus language. **R** is very widely used by professional statisticians as well as risk and quantitative modelling professionals in the investment and banking industry. Although it is a free and open source, it is at least as capable as S-Plus or similar enterprise strength solutions and supported by a large and growing library of reusable models useful for modelling financial instruments and portfolios.

To get your own free copy of **R** go to the companion web site http://modelbook.fincorefinancial.com/, where you will find up-to-date links and instructions to download and install **R** on your computer.

Free Companion Software: Personal ORACLE

All database examples (SQL and PL/SQL) in this book are written for use with ORACLE. ORACLE is the world's leading commercial database software. ORACLE provides a full industrial-strength free version of its software that you can download from their web site and install on your computer without charge. This free version is also optimised for use on your PC and is thus ideal for exploring all the SQL and database related code.

To get your own free copy of ORACLE go to the companion web site http://modelbook.fincorefinancial.com/, where you will find up-to-date links and instructions to download and install ORACLE on your computer.

Free Companion Software: AMPL

Some of the modelling examples in this book are written for use with AMPL. AMPL is one of the leading languages for mathematical modelling involving linear and nonlinear programming and constrained optimisation. If you want to run those examples in the book that use AMPL, or you want to write your own portfolio optimisation programs you should download and install AMPL on your computer.

To get your own free copy of AMPL go to the companion web site http://modelbook.fincorefinancial.com/, where you will find up-to-date links and instructions to download and install AMPL on your computer.

Build Instruments by Drag-and-Drop in Graphical Wizard

One of the best ways to explore the building block approach is by creating new instruments using a variety of building blocks from a rich palette of component blocks. The Fincore Investment Portal lets you create new financial instruments and templates for financial instrument types using an easy-to-use graphical wizard. With the trial version of the wizard you can create a very wide range of financial instruments using common and less common building blocks from coupon streams and bullet redemptions to exotic options.

To get a free trial use of the Fincore Investment Portal and the Fincore Instrument Builder Wizard go to the companion web site http://modelbook.fincorefinancial.com/, where you will get details for registering for a free trial account.

Acknowledgements

Writing this book was a great and fascinating challenge. It nevertheless required a lot of understanding and support from both my private and professional entourage.

In particular I would like to thank the staff at Elsevier: Karen Maloney, Stacey Walker, and Jeff Freeland, for their help and encouragement.

My first and last thoughts are always with my wife Susanne and our two children, Yule and Luka, who so marvellously supported me throughout the past year while writing this book. Thank you so much for everything!

Wolfgang Schwerdt
Versoix, April 2009

Writing a book is a monumental task not possible without the help and support of many people. I am therefore very grateful to all those who have given their help and support to Wolfgang Schwerdt and myself, thus helping to bring this book into the world.

In particular I would like to thank Karen Maloney, Stacey Walker, and Jeff Freeland from Elsevier for all their help, patience, and gentle but unyielding prodding and insistence for Wolfgang and myself to get the book completed.

Next I would like to thank Soeren Christensen, Predrag Popovic, and Chris Walker from Finsoft Ltd and Finsoft Financial Ltd for their support, both for the book itself and the product and project work at Finsoft Ltd and Finsoft Financial Ltd, which allowed me to put into practice many of the ideas and concepts in the book.

Particular thanks should go to Soeren Christensen, CEO of Finsoft Financial Ltd, who allowed us to use many examples derived from my work at Finsoft Ltd and Finsoft Financial Ltd.

I would also like to thank my colleagues at Finsoft and Finsoft Financial whom I had, and have, the pleasure to work.

Finally, last but certainly not least, I would like to thank my family for all their support, not just while writing the book but throughout my whole life. My mum

Franziska von Wendland and my dad Georg Kress have no small part in helping me become the person I am today and thus deserve special credit. I would also like to thank my brother Felix von Wendland and his partner Ana Abati for feeding and watering me and making me smile on more than one occasion while I was burning the midnight oil writing parts of the book.

Marcelle von Wendland
London, April 2009

CHAPTER 1

Introducing Model Implementation with the Building Block Method

This chapter introduces the building block approach. Any known financial instrument from the most simple to the most complex and esoteric can be put together from a small set of atomic components. With traditional data models you need to extend your data model every time you get a new unexpected combination of features. Then you need to develop new valuation and risk models for the new type of instrument. With the building block approach neither of the two steps are needed. All you need to do is to compose the new instrument from predefined building blocks that cover all imaginable features.

The fact that building blocks can be added easily and all instruments can be composed from a small set of building blocks means that a small number of building block valuation and risk models can handle an endless variety of actual financial instruments, rule sets can be built from simple components and configured to validate specific combinations of features dynamically, and data base, modelling, and portal software are not impacted by new instrument types but instead support them as soon as they arise.

1.1 Why Use a Building Block Approach?

Financial instruments come in a bewildering variety. There are virtually limitless possibilities to create new instruments by combining even just the more mainstream features in different combinations in order to match constantly changing market conditions and needs of investors. It is hence impossible to create a unique off-the-shelf valuation model for each combination of features used in the market. While most instruments are specimens of the pure plain vanilla variety of common instrument types like money market discount instruments, zero coupon bonds, fixed or variable coupon bonds, equity as well as derivatives make up a large fraction of all instruments in use at any point in time, this still leaves a large set of instruments that are more complex and often unique in one or more aspects.

However, help is at hand. Nearly two millennia before modern physics discovered the real atoms, Greek philosophers used the word *atomos*, meaning *that which is indivisible*, to describe the basic substance from which they thought all material things were constructed.

When you take on the task of valuing financial instruments with a comprehensive set of atomic instruments, it will allow you to split up more complex investments into combinations of simpler ones. This means that you will need a much smaller set of tools than you would have otherwise.

Robust and sophisticated tools whose behaviour is well known can be constructed to allow you to determine the value, risk, and profitability of the atomic parts of any instrument and ultimately that of the instrument of which they form a part. In addition, the smaller set of tools will be more manageable to learn and master.

Figure 1.1 illustrates how you can quickly build actual financial instruments by dragging common building blocks from a palette onto the instrument definition. In the example a Fixed Coupon Bond is composed from two building blocks: a Fixed Coupon and a Bullet Style Fixed Redemption building block. In Figure 1.2 you can see the details for the Fixed Coupon building block of the example Fixed Coupon Bond.

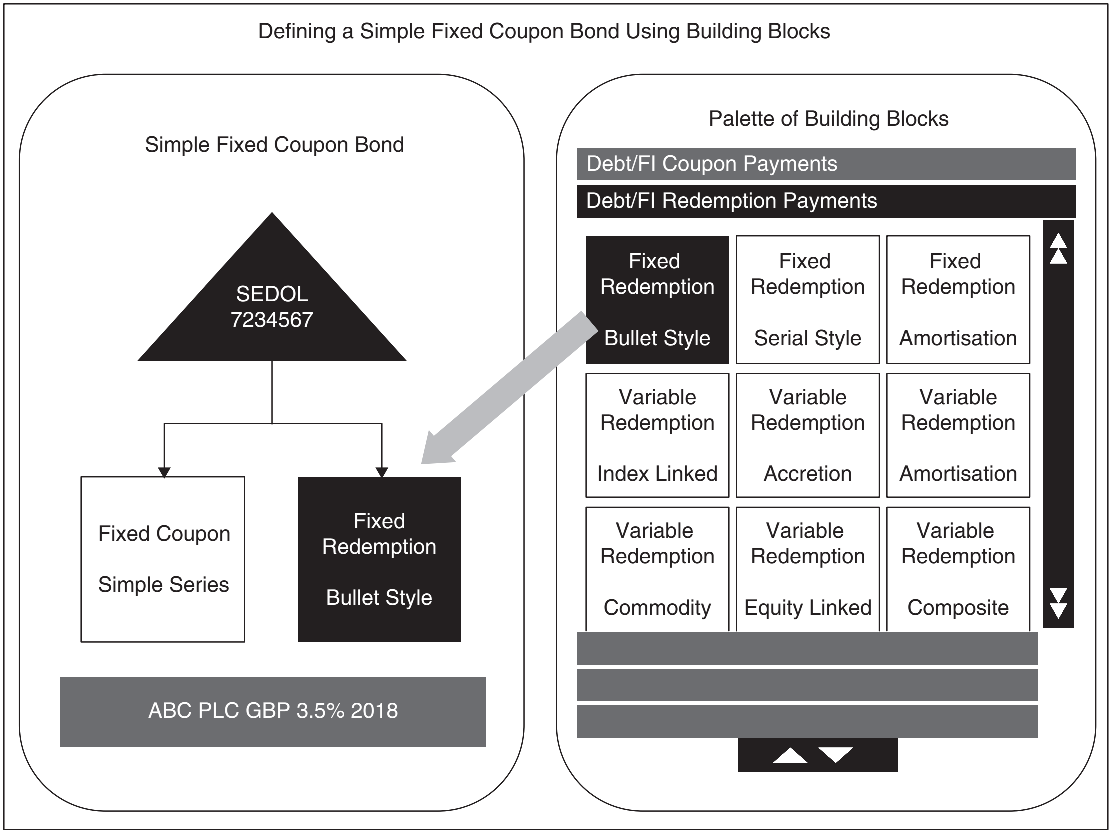


Figure 1.1: Composing a financial instrument from a palette of building blocks.

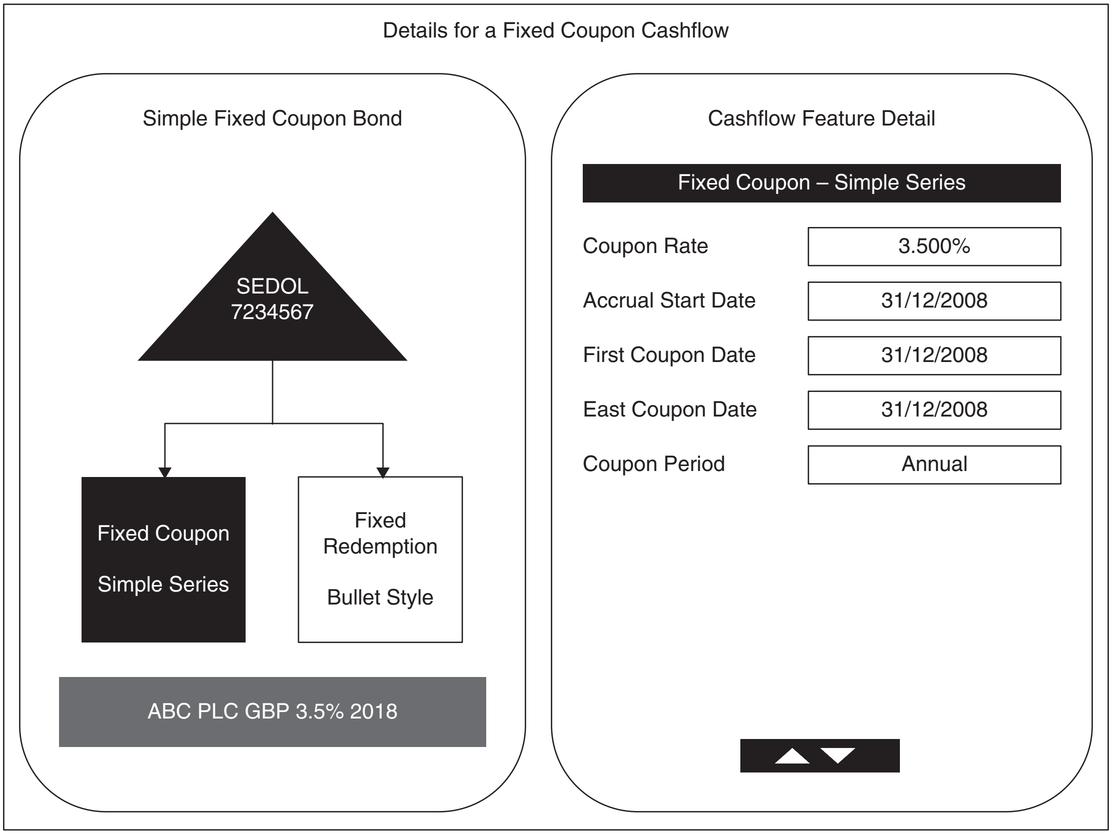


Figure 1.2: Completing and viewing the details of a component building block.

Fixed Coupon Bonds of course are handled easily by more conventional models, but there are many structured products that combine endless combinations of different plain vanilla and exotic features into different unique products. One of the big attractions of the building block approach is that you can easily create instruments of any complexity using a graphical drag-and-drop wizard tool (such as the one on the companion web site).

This is where the building block approach starts to pay off. With traditional data models or approaches you would need to extend your data model every time you get or want to use a new unexpected combination of features. You would now need to develop new valuation and risk models for the new type of instrument. With the building block approach neither of the two steps are needed. All you need to do is to compose the new instrument from predefined building blocks that cover all the separate features.

Very occasionally a new building block will be needed but this usually can be added without any data model changes again since the underlying data frame for all building blocks—called the Cash Flow Element—already has a super set of attributes that allows the definition of a near limitless variety of new building block types.

The fact that building blocks can be added easily and all instruments can be composed from a small set of building blocks not only allows very intuitive user interfaces for the creation of new instruments such as the Graphical Instrument Wizard from Fincore Financial, but it also means that a small number of building block valuation and risk models can handle an endless variety of actual financial instruments. Rule sets can be built from simple components and configured to validate specific combinations of features dynamically. Data base, modelling, and portal software is not impacted by new types of instruments but instead supports them as soon as they arise.

1.2 An Implementation Framework

Implementing valuation, risk, and performance measurement models is a complex task even in the simplest cases. Fortunately the task can be broken down into many smaller steps that are much easier to complete. To do this successfully you will need a framework and methodology to hold all the smaller steps together. In this chapter we introduce such a framework. This framework has been derived from a framework set out in Innmon, Zachman, and Geiger, 1997. In Figure 1.3 we outline the model implementation framework. Figure 1.4 provides an overview diagram for a generic methodology for implementing models.

In the remainder of this chapter we will look at how to apply the implementation framework and methodology. Because implementing models is such a complex endeavour, only the most simple ones can be implemented in one go from the first identification of a need right through to ongoing use of the model for decision making.

As a first step we will therefore look at the role of iterative cycles in the implementation process. The pattern we suggest to use is the iterative fountain model with short parallel planning and delivery cycles. Then we will look at each perspective in the framework and methodology starting with the **methodology perspective**. Although it might appear surprising to cover the task of methodology definition in an iterative and practical approach, there are good reasons to do so. The usefulness and reliability of risk modelling efforts in a particular context are highly dependent on the way the models are developed, implemented, and operated. What may have worked well in one context may well prove of little value or downright dangerous in another.

Next we will look at the **requirements perspective**. It may seem an obvious step but it is often underestimated. The results of modelling are highly dependent on the question we try to answer. It is important to ensure that the questions we try to answer with a set of risk

Model Implementation Framework							
Perspective	Goals Dimension	Organisation Dimension	Schedule Dimension	Model Dimension	Data Dimension	Process Dimension	Role
Methodology Perspective	Goal Setting Guidelines	Organisation Guidelines	Schedule Guidelines	Modelling Guidelines	Data Guidelines	Process Guidelines	Business Governance
Business Perspective	Business Case/Plan	Organisation Requirements	Schedule Requirements	Model Requirements	Data Requirements	Process Requirements	Business Owner
Architecture Perspective	Architecture Plan	Organisation Specifications	Design Schedule	Model Specifications	Data Specifications	Process Specifications	Model Designer
Systems Perspective	Build Plan	Responsibility Model	Build Schedule	Model Designs	Msg & Data Store Design	Process Designs	Model Builder
Operating Perspective	Operating Plan & Targets	Responsibility Assignments	Maintenance Schedule	Production Models	Production Data Stores	Operating Schedules	Model Operator
	Why?	Who?	When?	How?	What?	Where?	

Figure 1.3: Model implementation framework.

Model Implementation Methodology Template							
Set of Deliverables	Goals	Organisation	Schedule	Model	Data	Process	Persons Responsible
Methodology	Develop and Implement Goal Setting Guidelines	Develop and Implement Organisation Guidelines	Develop and Implement Scheduling Guidelines	Develop and Implement Modelling Guidelines	Develop and Implement Data Guidelines	Develop and Implement Process Guidelines	Governance Board
Requirements	Develop and Implement Business Case & Plan	Develop and Implement Organisation Requirements	Develop and Implement Schedule Requirements	Develop and Implement Model Requirements	Develop and Implement Data Requirements	Develop and Implement Process Requirements	Business Owner
Architecture	Develop and Implement Architecture Plan	Develop and Implement Organisation Specifications	Develop and Implement Design Schedule	Develop and Implement Model Specifications	Develop and Implement Data Specifications	Develop and Implement Process Specifications	Model Designer
System	Develop and Implement Build Plan	Develop and Implement Responsibility Model	Develop and Implement Build Schedule	Develop and Implement Model Designs	Develop and Implement Msg & Data Store Design	Develop and Implement Process Designs	Model Builder
Operations	Develop and Implement Operating Plan & Targets	Develop and Implement Responsibility Assignments	Develop and Implement Maintenance Schedule	Develop and Implement Production Models	Develop and Implement Production Data Stores	Develop and Implement Operating Schedules	Model Operator
	Why?	Who?	When?	How?	What?	Where?	

Figure 1.4: Model implementation methodology template.

models are really the ones that need to be answered to make the right decisions for a successful outcome from a business perspective.

Following the requirements perspective comes the **architecture perspective**. Any nontrivial risk model implementation is a complex (software) engineering task and will produce a nontrivial software (engineering) system as a result. To ensure that the overall system will fulfil the expectations placed on it when it is first delivered, and likewise years later after accommodating many changes, it is vital that it is built on the basis of blueprints that ensure its overall efficiency and effectiveness as much as its resilience in the face of change.

The **system perspective** will complete the sequence of steps necessary for the development of particular implementations of risk models. This is the sharp end where we need to make decisions about all the details that determine the best way to realise risk models as working software components. Beyond the models themselves we also need to ensure we have or create the right components that feed our models with the right input at the right time, and allow us to distribute and analyse our modelling results to support the decision-making processes for which we built the models in the first place.

Finally we will cover the **operations perspective**. Again this may be a surprise since a model is operated only once its development has been completed. However as we initially noted, only very few models can be successfully implemented in a single go without the need to elaborate them or at least adapt them in the wake of change in the environment and user needs. In this logic the operations phase becomes an integral part of the model development lifecycle.

1.2.1 The Iterative Process of Implementing Risk Models

Implementing risk models requires many decisions and assumptions whose impact can usually be assesed only by reviewing the perfomance of the resulting implementation in practical use. This makes an iterative implementation approach a necessity for all but the most trivial projects.

The iterative approach works best with short, well-defined iteration cycles such as one month. Weekly cycles also may be appropriate in some cases, and maybe six or even eight weeks in others. Shorter cycles tend to work better if the major focus is on modelling itself, whereas slightly longer cycles may be more appropriate to very large-scale projects focussed predominantly on software development where the actual models are already well known, tried and tested for the given circumstances.

The iteration approach we recommend as part of the Building Block implementation methodology is illustrated in Figure 1.5. It has two parallel streams for each cycle and a

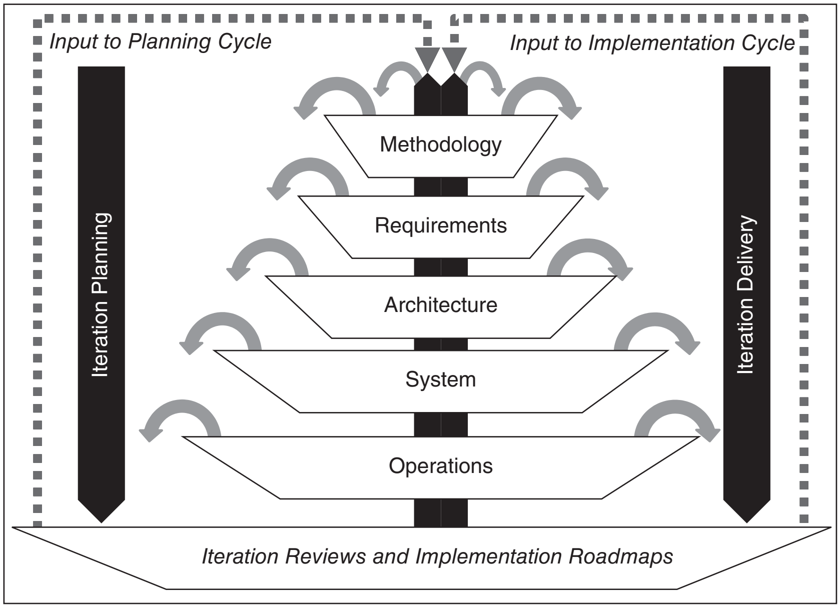


Figure 1.5: The Fountain model for iterative implementations.

review between cycles: The first stream, *Planning*, is concerned with preparing the ground for the implementation work in the following iteration.

The second stream, *Delivery*, is concerned with creating the deliverables that are needed to realise the models both as working software but also as real-life business processes delivering the required results in a given organisational context.

Every time one iteration ends and another one starts it is also very beneficial to review the lessons from the iteration just finished and feed it back into the forthcoming iterations.

Let us now briefly review the role of each of the three elements of the iteration approach in the context of model implementations.

1.2.1.1 Iteration Planning Cycle

With risk model implementations numerous problems will arise and have an impact also via interdependencies at many different levels, from governance all the way to day-to-day operations. Making changes in response to perceived gaps or issues often requires input from people having different perspectives, different backgrounds and qualifications in order to tackle the same gap or issue at different levels of our implementation framework.

The purpose of the planning cycle is to give time and provide a structure for the vital dialogue that needs to occur before any commitments are to be made to address a gap or issue in a particular way.

The iteration planning cycle is therefore not a project management activity focussed on crafting schedules or waterfall-type paint-by-number dead specifications. It is instead a process that aims at efficiently tackling gaps or issues through a structured dialogue. This dialogue gives relevant participants from each of the five perspectives the opportunity to contribute.

Iteration Planning should not be a weighty or bureaucratic process. Often it is best done through joint review workshops carried out after participants have had a chance to review the original gap or issue in writing and put forward brief written comments setting out alternative solutions or highlighting interactions or constraints.

At the end of the dialogue each gap or issue should either have an agreed solution or should be referred to a future iteration. The solutions should ideally be small enough to be developed or otherwise created in less than a week by one or two participants, but this may well vary somewhat depending on circumstances. Before the solutions are handed over to the delivery cycle they should also be prioritised and fitted into a suitable place in a roadmap so that it is clear what needs to be done now and what later, and which things need to be carried out together.

1.2.1.2 Iteration Delivery Cycle

The iteration delivery cycle is the workhorse of model implementation. If the planning cycle is working well the planning activities will leave the majority of time of all participants to be devoted to the delivery cycle itself. If the solutions, priorities, and the roadmap of the planning cycle are “fit for purpose”, then project participants can focus on implementing as many solutions as possible in priority order without much additional overhead to coordinate the delivery work any further.

Model implementation involves many different qualifications. By splitting the solution dialogue and coordination effort from the actual delivery work itself, it becomes possible to maintain communication overhead at a minimum while ensuring that necessary communication does really take place and dialogue time really contributes to better solutions.

1.2.1.3 Iteration Reviews and Roadmaps

Iteration Reviews and maintaining an implementation roadmap are not a separate stream but should be done as part of the planning cycle. They are, however, a key backbone that ties the planning and delivery cycles together.

Iteration reviews are best built up by recording gaps, issues, and observations as they occur during delivery and then feeding them into the planning cycle after they have been reviewed by participants and any observations or responses have been recorded by participants.

At the other end of the planning cycle we should see a clear roadmap emerge that ties together different solutions into logical groups that should be or must be delivered together. It further visualises the priorities that are assigned to different solutions.

The roadmap allows the delivery work to proceed at an optimal pace and provides, together with the solution description, the information needed to create quality results delivered when they can make the biggest possible contribution.

1.2.2 Modelling Methodology Perspective

Model implementation and ongoing model operations almost always are and should be deeply woven into an organisation's fabric. This means that the goals and schedules need to be aligned with an organisation's wider goals and schedules with respect to risk management, including those set externally by regulators or voluntary codes.

The organisation of risk modelling projects and ongoing operations also needs to be aligned to both the complexity of the task itself and the organisational, regulatory or governance framework in which they take place.

Finally the selection of models, data, and processes also needs to fit both regulatory and governance constraints and be in line with both the organisation's actual capabilities and the organisation's tolerance to errors in any particular risk modelling efforts.

At the risk modelling methodology or governance level you therefore need to address this alignment between your project and the wider organisation. As you plan your next iteration or the roadmap beyond, you will need to bring in the necessary expertise and involve the right stakeholders to ensure that the alignment with the wider organisation and beyond can take place.

Thus, you should include not only the key project participants and the business sponsor for the project, but also any additional experts as well as other key stakeholders such as those with a governance responsibility that includes the business units directly involved in the project.

As part of the planning cycle, at this level you should agree on any modifications to the overall planning and delivery approach for the forthcoming planning and delivery cycles, and, where possible, for stages of the roadmap beyond the current iteration cycles.

Risk modelling projects, because of their complexity, rely strongly on an appropriate approach for good results. What is appropriate can vary tremendously over time and in different situations. It is therefore important to check in each iteration whether the approach needs to be adapted and, if so, ensure that the right modifications are made and actually work.

This helps to break down the methodology and guidelines into the different perspectives: Goals, Organisation, Schedules, Models, Data, Process.

1.2.3 Modelling Requirements Perspective

Risk Models are never created or implemented just for their own sake. It is therefore paramount to establish the true Modelling Requirements for the project from a business perspective before each iteration. If a project is broken down into sufficiently small and frequent iterations then the requirements can be refined and elaborated from iteration to iteration. The modelling effort from earlier iterations will help throw light on constraints such as the available experience of model developers and model operators; the availability, reliability, and timeliness of required data; the performance and accuracy of models, and so on.

It helps breaking down the requirements into different perspectives: Goals, Organisation, Schedules, Models, Data, and Process. Let us now look at Modelling Requirements from each of the six perspectives.

GOALS: Why Are We Really Doing This?

The first perspective to consider should be the ultimate goals of the project from a business perspective, because these requirements are fundamental to all the others. Goal requirements are concerned with the question, "why are we doing this?"

It is very tempting to answer this question with little introspection and just repeat whatever may have been the apparent trigger for the project. With a risk management project this is particularly dangerous as it leaves a large number of assumptions that will have to be made both unarticulated and untested.

Not properly clarifying the underlying goals of the project will vastly increase the potential for misunderstanding both within the project and between the project and external stakeholders. It will also make it difficult or even impossible to check whether the project is delivering results that are fit for purpose or not.

The question you should consider is "why are we *really* doing this?" Thus a goal such as "we need to know our VAR / ETL at 99%" should be carefully investigated and if left standing only be there to supplement the real reasons. The kind of questions you should ask to get to the real requirements are, who will be using the information, for what purpose, and in what context?

Once you have elaborated the real goals sufficiently it is also important to consider if the set of goals is consistent and achievable.

ORGANISATION: Who Will Need to Be Involved?

The next perspective is to look at the organisation of the project; in other words, who will need to be involved, and how, in order to

- Contribute from a governance point of view
- Contribute knowledge and expertise
- Plan and execute the project
- Implement the models and supporting data and process infrastructure
- Supply the inputs and infrastructure for the models
- Operate the resulting models
- Use the information generated by the models once in operation

In Risk Modelling projects the significance of who needs to be involved and how is of high significance. The complexity of the modelling task and the difficulty of visualising the fit of the models as implemented, with the needs and constraints of those involved once they are operational, makes it imperative that you have a robust plan for who to involve and how.

SCHEDULES: When Will We Need to Reach Certain Milestones?

Model implementations are usually tightly linked to other projects and organisational objectives. It is therefore imperative that you get a clear view of the interdependencies and agree on a realistic roadmap flexible enough to accommodate inevitable changes.

MODELS: Are There Constraints on How We Can Model Risks in the Project?

Even the most ambitious well-endowed greenfield project will face a range of constraints on which models can be used and how they can be implemented. The constraints come from a variety of sources such as regulatory requirements and your organisation's capabilities and resources.

DATA: What Inputs Do We Have Available for Driving the Models?

Data is a key aspect when planning to implement models. Even the most sophisticated and robust model will produce unreasonable results if fed with corrupt or wrong information, and will stop working altogether if certain inputs are unavailable. A realistic roadmap for what data will be available, and when, is critical for a successful implementation. Often an implementation must devote considerable efforts to obtaining data hitherto unavailable and you need to factor this into your plans.

PROCESS: Where and With Which Resources Will the Models Be Operating?

This final dimension must take into account both project resources and the resources that will be available for operating the models you deliver once they have moved into production. It is no good to deliver an ideal model that trumps other alternatives but requires twice as many resources to run on an ongoing basis than what the user of your model results can afford to pay. It is critical to your success that you know as early as possible when there is a conflict between what is affordable and what is required in terms of ultimate goals.

1.2.4 Modelling Architecture, Systems, and Operations Perspectives

The architecture, systems, and operations perspectives again cover the dimensions: Goals, Organisation, Schedules, Models, Data, and Process. Each is a gradual refinement and stepwise instantiation of the requirements within the bounds of the methodology. You will find more information about these perspectives on the companion web site (http://modelbook.bancstreet.com/).

To help you work through each of those perspectives, go to the companion web site and work through Lab Exercise 1.1.

Lab Exercise 1.1: Implementation Planning

1. Download the file **ImplementationPlanning.pdf** and work through the step-by-step planning tutorial for either a fictitious or real project within your organisation.

This page intentionally left blank

CHAPTER 2

Introducing the Building Block Data Model

This chapter introduces a data model that we believe is reasonably robust even for the most involved applications. The version presented here is largely sufficient to support any analytics discussed in this book. The data model has the advantage of (1) being able to dynamically store and integrate new information items that are not foreseen at system design time, and (2) being instrument type agnostic; that is, you can store information on all imaginable financial instruments in one single framework.

The pricing, risk, and performance measurement algorithms presented in this book can also be implemented using spreadsheet-style data structures. One can make the bridge between the uni- or two-dimensional data requirements of the measures and the multidimensional data model by introducing so-called "views," which are spreadsheet-style data structures extracted from the relational data model via an intermediary SQL statement.

2.1 Considerations When Modelling Financial Data

Before you start building your application you need to have a good grasp of the data you want to process, both in conceptual and physical terms. In this book we focus on the conceptual data model. However the practical aspect of data handling, such as capture, storage, and data quality management, should not pass unnoticed. They are essential for your system to be successful and hence will be briefly discussed in Section 2.3.

There are different philosophies on how to best model financial data and, in a way, it depends on the purpose of the application you are about to develop. However, once you plan to develop anything more integrated than what can be done in a simple spreadsheet, you will have to consider relational data modelling and storage.[1] There are simply too many time-varying aspects, repetitive structures, and interrelationships around financial instruments than would properly fit into a simple spreadsheet.

In this chapter, we introduce a simplified version of a data model that we used in practice and that we believe is reasonably robust even for the most involved applications. The

[1] See Date, C.J. (2005; 2008) and Simsion and Witt (2004) for in-depth discussions of relational database theory and application.

version presented here however is largely sufficient to support any analytics discussed in this book. The data model has the advantage of being

- Able to dynamically store and integrate new information items that are not foreseen at system design time
- Instrument type agnostic; you can store information on all imaginable financial instruments in one single framework

The pricing, risk, and performance measurement algorithms presented in this book can also be applied using spreadsheet-style data structures. However, because each type of measure is based on a different dimension of financial instruments (prices, classifications, planned or actual cash flows), building an application to encompass all types involves a data structure capable of supporting multidimensional, hence relational, data items.

One can make the bridge between the uni- or two-dimensional data requirements of the measures and the multidimensional data model by introducing so-called "views," which are spreadsheet-style data structures extracted from the full data model via an intermediary SQL statement. So, for a specific application, only a subset of the data can be represented in a simple spreadsheet. We will not bother you with the (sometimes quite involved) SQL algorithms creating the views from the relational database.

2.1.1 What Is Important to Store in Financial Instruments?

Before we move on to the data model as such, it is useful to shortly recall what we exactly are about to model: the world of financial instruments. More precisely, which aspects of financial instruments are relevant for storage in a database. From a bird's eye perspective, financial instruments are

- Legal contracts between a creditor and a debtor. The creditor is the one who pays money to the debtor in the hope for a financial return. In this book, we call that role the **investor**. The debtor is the one receiving the money, which he or she invests in some business of his or her own, hoping that the investment will generate more cash than the money paid to the investor. The debtor is called the **issuer** of the financial instrument.
- Purely about the exchange of money. They do not directly involve any tradable goods or services, hence the term "financial."[2]

[2] Apparent contradictions to this rule could be Repurchase Agreements or Collateralised Debt instruments (such as CDOs, ABS, MBS, etc.). However, Repurchase Agreements are simple agreements on other financial contracts (securities), and collateralisation is a guaranteeing device trying to reduce the risk of loss for the debtor, not the first objective of the financial contract.

Seen from that angle, a financial instrument essentially defines a series of cash flows or optional claims on cash flows (options) that are (to be) exchanged between the investor and the issuer. This series, its timing and conditioning, is called the **cash flow structure**, and is the central feature of any financial instrument.

Financial instruments give the right to receive (or the obligation to provide) monetary cash flows. Thus, the acquisition of a financial asset can be summarised in terms of a sequence of monetary cash flows, including the purchasing price. This definition immediately leads to the prime feature of financial instruments, being claims on a series of future cash flows. Thus, from a valuation perspective, their prime characteristics are their

- Cash flow structure
- Risk structure of the implied cash flows

The promise or hope for cash flows is the *raison d'être* of financial instruments: The issuer "borrows" the principal amount, invests it into business, and makes the investor participate in the wealth generated from the business by delivering income payments. In the case of debt or closed fund shares, the issuer repays the money borrowed upon maturity of the instrument.

In its most simple form, the investor pays a lump-sum (also called principal or nominal value, depending on the instrument) to the issuer who in exchange promises (a series) of cash flows and, in many cases, the ultimate repayment (redemption) of that lump-sum.

Figure 2.1 shows a schematic view of this basic cash flow structure. The dotted arrow from the issuer to the investor highlights the fundamental uncertainty inherent to the repayment of the money paid out by the investor.

It is this uncertainty that leads to quite a number of enhancing legal and other guaranteeing provisions (such as collateral allocation), with the aim to reassure the investor that he or she will actually get his or her money back. Such provisions may be conformance with regulatory requirements on the side of the issuer, a guarantee commitment by a third party, or creditworthiness assessments (ratings) of the issuer or the instrument itself.[3]

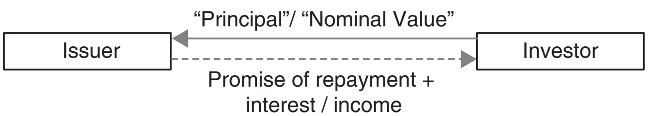


Figure 2.1: Schematic view of a simple cash flow structure.

[3] Creditworthiness measures the likelihood that a debtor (issuer) will repay his debt (the principal/nominal value). It may hence be surprising that an instrument may have a creditworthiness rating. This is possible because of a direct linkage of underlying cash flows that the issuer has agreed with a third party and legally fully attributed for serving the cash flow obligations of the instrument in question.

By varying the basic cash flow structure outlined in Figure 2.1 on the principal, income, redemption (repayment) modalities, or optional claims, financial engineers can and do create more complicated cash flow structures in practice. We will discuss the most predominant cash flow structures and their representation in our data model in Chapters 3 and 5.

A direct contract conclusion between the issuer and the investor is called **primary market** activity. Certain instruments, which follow a standardised format, can be resold by the investor to a third-party investor.[4] Such reselling activity is called the **secondary market**. Here, it is important to understand that the new or third-party investor does not "buy" the financial contract. He rather buys the right to enter as a new investor the existing standardised contract with the same issuer.

2.1.2 Overview of the Conceptual Model Areas

With this abstract but powerful view of financial instruments we can now proceed with identifying the main conceptual areas that need to be captured by our data model. If you are already familiar with financial instrument modelling you will quickly notice an important feature of the model we present here: It is instrument-type agnostic!

In contrast to most financial data models we do not have any attribute classes that apply only to a certain type of instrument, such as stocks, bonds, or derivatives. Based on this logic any, however complicated, financial instrument can be represented by a series of elementary instruments (building block principle), each representing a relatively simple cash flow structure. We focus on modelling (and then combining) these elementary instruments rather than creating ever more complex and difficult-to-maintain data structures. We even use this approach to model portfolios of instruments, since, logically, they are just an extension of the building block principle.

The trade-off of this approach is an increased modelling and mapping effort, and a data model that, to the uninitiated, is less intuitive than conventional data models. The big gain, however, is that you have nearly unlimited modelling options and are very unlikely to have to adapt your data model in the foreseeable future. Because any change to the data model means the same to a system as does a heart chirurgical intervention to a human, we believe this approach to be justified and highly beneficial.

We will give in-depth details on how to use the data model to represent financial instruments in Chapters 3 and 5. But before that, let us return to building up our logic and find out what actually needs to be modelled and where:

[4] Typical examples are instruments registered at a regulated stock exchange, called **securities**. Securities have to follow a format prescribed by the stock exchange management, aimed at enhancing the transparency with respect to the instrument and its issuer to all potential investors, making price finding as efficient and fair as possible.

- **Instrument Core**: Every financial instrument has some fixed contractual elements that never change during the life of the instrument. Examples are the denomination of the instrument (nominal value), the currency of the denomination (nominal currency), among others. This core information cannot be allowed to change, as it would make the instrument for both the issuer and investor an uncontrollably hazardous experience.
- **Cash Flow Structure**: Concerning the cash flows (to be) exchanged, three classes may be distinguished:
  - **Scheduled Cash Flows**: These are promised at contract conclusion (issuance) and hence are known throughout the life of the instrument. In the case of time varying cash flows, we need to know how these are to be calculated; that is, which formula based on which **underlying** is to be used in which transformation.
  - **Fixed Cash Flows**: In case the cash flow schedule does not refer to a specific series of amounts, but rather to a rule, determining the amount to be paid at a certain date in the future, these amounts are being fixed a predefined intervals before their actual payment date.
  - **Actual Cash Flows**: Instances of actually paid cash, like coupons or dividends. These are particularly relevant when no cash flow schedule is promised at issuance, or if the cash flow schedule is a time varying one.
- **Rating**: The correct discounting of (scheduled) expected future cash flows requires the mapping of an appropriate discount curve to the instrument. This is done by allocating curves representing the same creditworthiness/default risk as the issuer.
- **Industry Classification**: When pricing an instrument via relative pricing or measuring its performance, the mapping of an appropriate industry benchmark index is key. For this, we need to know the industrial sector of the instrument.
- **Market Data**: Real instrument prices and their provenience (stock exchange, data source estimate, multilateral trading platform, etc.) are key to most parts of this book.
- **Issuer Party**: Information on the instrument issuer, notably its credit rating.
- **Instrument Portfolios**: Sets of financial instruments.

2.1.3 The Conceptual Data Model

When designing a database, you normally start at the conceptual level. Here, you do not need to consider the details of actual physical implementation. A conceptual data model represents the overall logical structure of a database, which is independent of any software or data storage structure. A conceptual model gives a formal representation of the data needed to run an enterprise or a business activity. It fulfils the following roles:

- It represents the organization of data in a graphic format.
- It allows verification of the validity of data design.
- It allows generation of a Physical Data Model, which specifies the physical implementation of the database.

The data model is made up of attribute classes (each corresponding to a physical database table) that are linked by one-to-one, one-to-many, or many-to-many relationships. Each class represents a distinct real-world aspect of financial instruments.

As our data model is instrument-type agnostic, all attribute classes in principle are applicable to all types of financial instruments. This allows us to limit software reengineering in case the system needs to support new classes of financial instruments, including financial innovation.[5]

The conceptual areas of the previous section may be captured by the structure outlined in Figure 2.2. Attribute classes here are grouped into logical submodels, each assembling attribute classes describing one of the major entities in the life of a financial instrument:

- The **Instrument Model** describes the instrument itself. It captures all information not related to parties and markets (such as prices).
- The **Portfolio Model** describes the tables that are used to set up and store information on instrument portfolios.
- The **Party Model** describes the characteristics of the real world legal and physical entities playing a role in the life of a financial instrument.
- The **Role Model** describes any relationship between an Instrument and a Party.
- The **Market Model** is a listing of instruments on regulated markets (e.g., stock exchanges) and actual instrument prices; that is, valuations of financial instruments by market participants. The Market Model makes references to both the Instrument and the Party Models.

[5] By breaking down the structure of financial instruments into defined logical blocks, the data model is capable of supporting a nearly unlimited set of specifications, in particular derivative products and new, more complex instrument structures (financial innovation).

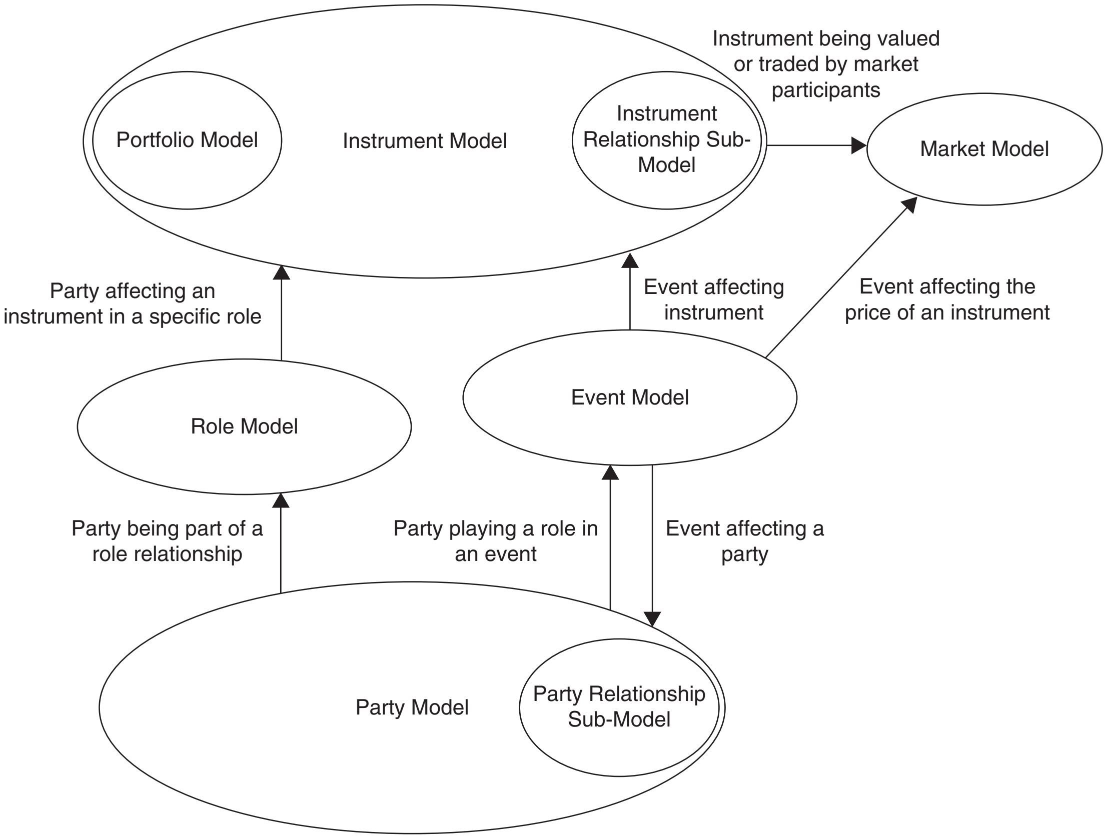


Figure 2.2: Full logical structure of the conceptual data model.

Table 2.1: Stylised attribute class table

Attribute Name	FK/PK	Description	Type	M/O	Domain

2.1.4 Modelling Conventions

The attribute classes physically represent tables the structural details of which are presented in the appendix (see Table 2.1). The structure of the tables is as follows:

- **Attribute Name**: Name of the attribute in business terms. In our SQL code examples, the spaces between the words are replaced by underscores "_". For example *Instrument ID* becomes *Instrument_ID* or *Nominal Currency* becomes *Nominal_Currency*.

- **PK/FK**: Indicates whether the attribute is the *Primary Key* (PK) or a *Foreign Key* (FK) of the table. Together, these keys allow a unique identification of every entry in the table.
- **Description**: Business content of the attribute.
- **Type**: Abstract data type of the attribute (see later).
- **M/O**: Indicates whether filling in an attribute is mandatory (M) or optional (O).
- **Domain**: Domain of the attribute; that is, the set of values that are valid for the attribute.

In the conceptual data model we use abstract data types. Which exact physical data types to use (e.g., integer, double, etc. for the abstract types "Number" or "Amount") depends on the software environment (database, programming language), storage capabilities, and computation power. Table 2.2 gives an overview of the abstract data types we use in this chapter.

Although we use the abstract data types in the conceptual model, you will find concrete data type specifications in the SQL sample code given for each table later. Although we consider these to be reasonable starting values for setting up your database, we nevertheless recommend that you verify their appropriateness with your database administrator, to accommodate system-specific restrictions or performance enhancement options such as index generation.

We further model code lists as simple look-up tables (value lists). Properly engineered, they should also represent a set of classes that are linked to the main attribute classes of the model via internal IDs.

Table 2.2: Abstract data types of the conceptual data model

Type	Description	Example
Identifier	System identifier. When being a Primary Key attribute, the value is auto-generated by the system.	4524856
Look-up	Set of values specified in a code lists table.	WEEKLY, DAILY
Indicator	Boolean value (Yes, No) or (–1.0)	–1, 0
Date	Date & time; the precision required depends on the attribute.	21.07.2008
String	Text	Univalue Funds
Amount	Monetary amount	17.28
Rate	Rational number	4.5000
Number	Large number	1,500,258.00

In the following sections we present the tables in technical detail, after each table giving the SQL code to create it.

2.2 The Data Model

2.2.1 The Instrument Model

The Instrument Model is centred on the *Instrument Core* class. Linked to this class are groups of logically related attribute classes (tables) that store the information describing the various logical areas supported by the model.

Figure 2.3 shows the full conceptual structure of the Instrument Model. The main topical areas are Issuance Details, Relationships to other Instruments, Qualitative Information such as classifications or credit ratings, Market Information, Analytical Information, and Cash Flow Information (scheduled and actual).

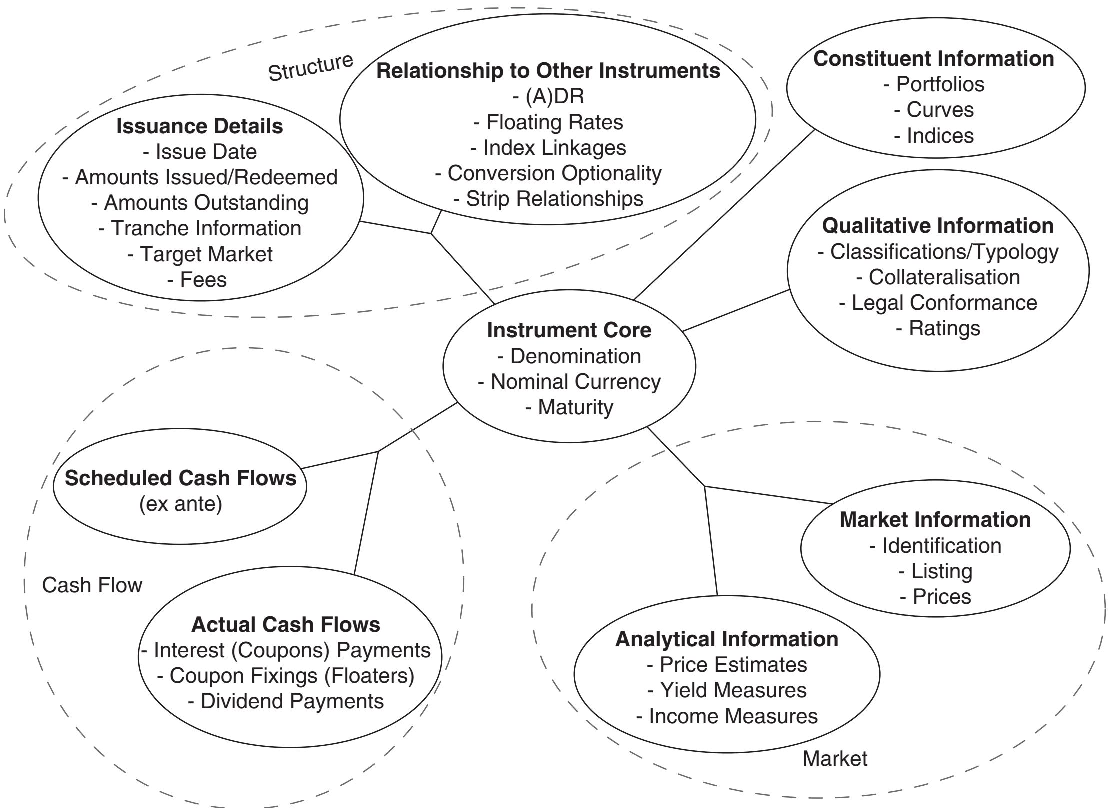


Figure 2.3: Logical structure of the full instrument model.

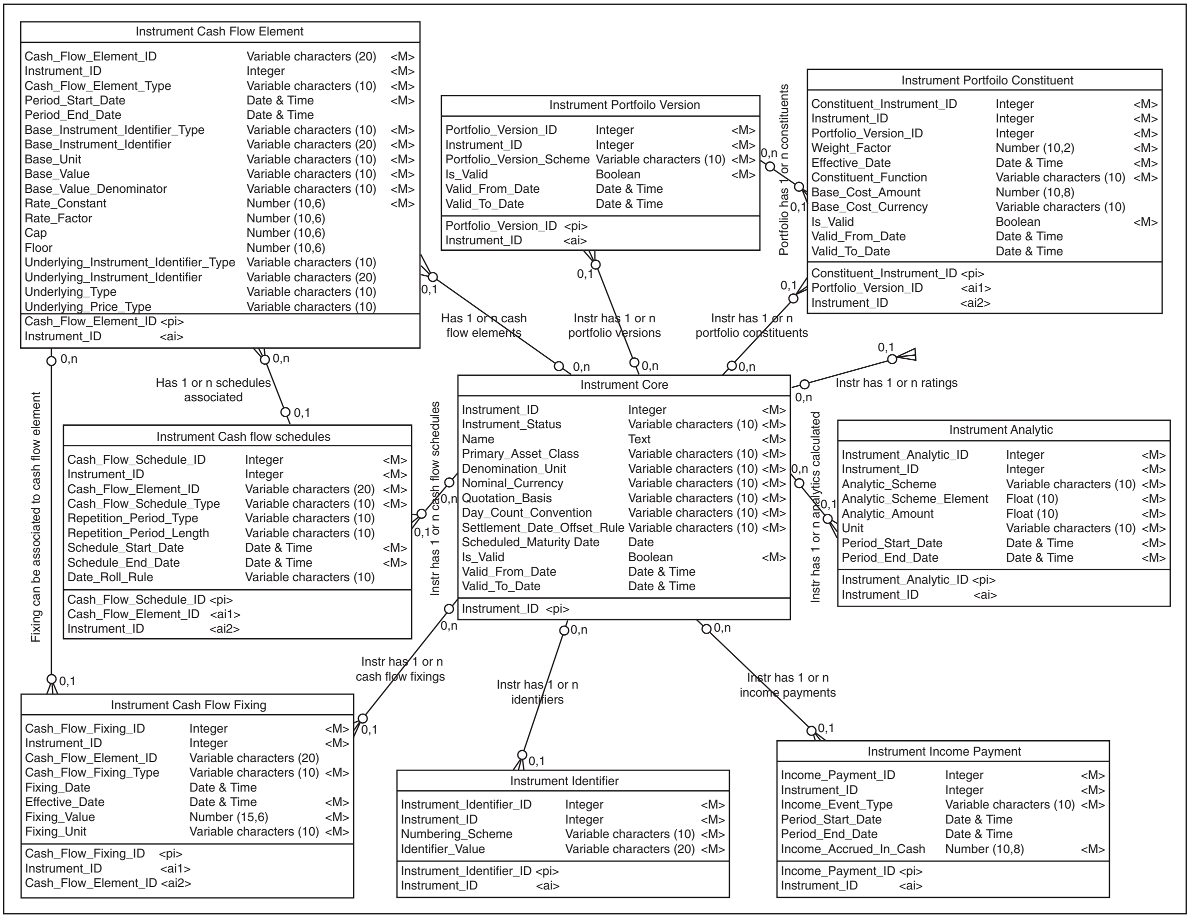


Figure 2.4: Conceptual view of the instrument data model classes.

In this book, we do not work out all the attribute classes necessary to store all that information. We rather content ourselves to those classes needed to reproduce the calculations presented in this book.

Figure 2.4 gives an overview of the classes of the instrument model that we use in this book and that are presented in the following sections.

2.2.1.1 Table: Instrument Core

The *Instrument Core* class of attributes stores information that is generic to all financial instruments. It also provides the central link to the other attribute classes that describe more specific aspects of a financial instrument. See the appendix for the full structural table description.

2.2.1.2 Table: Instrument Analytic

The *Instrument Analytic* class allows representing analytic values such as estimated prices (as far as not stored in the *Instrument Price* class), accrued income, and other calculation results like Beta, Yield, and Volatility. It is the place where you store calculation results for later reuse.

See the appendix for the full structural table description.

2.2.1.3 Table: Instrument Cash Flow Element

The *Instrument Cash Flow Element* class allows representing the scheduled cash flow structure of the instrument; for example, any fixed, optional, or residual income or capital redemption cash flows scheduled for the instrument. The *Instrument Cash Flow Element* class is used in conjunction with one or more instances of the *Instrument Cash Flow Schedule* class. The details of modelling scheduled cash flow structures are discussed in more detail in Chapters 3 and 5.

See the appendix for the full structural table description.

2.2.1.4 Table: Instrument Cash Flow Schedule

The Instrument *Cash Flow Schedule* class specifies the first and last dates on which a particular cash flow event for a Cash Flow Element is planned to occur, as well as how to determine any future regular recurrences of that event.

See the appendix for the full structural table description.

2.2.1.5 Table: Instrument Cash Flow Fixing

The *Instrument Cash Flow Fixing* class of attributes allows representation of actual fixings that have taken place defining the actual values of any variable coupon, redemption, or cash flow related parameters such as pool factors for any instrument.

See the appendix for the full structural table description.

2.2.1.6 Table: Instrument Identifier

The *Instrument Identifier* class of attributes captures information on the instrument identification. Data on multiple identifiers for the same instrument can be stored in a single, repetitive database table.

See the appendix for the full structural table description.

2.2.1.7 Table: Instrument Income Payment

The *Instrument Income Payment* class records the history of actual income payments for an instrument. This includes declared and actually settled dividend payments as well as actual coupon payments. For the purpose of this book, we abstract from noncash income as well as any tax considerations.

2.2.1.8 Table: Instrument Index Version

The *Instrument Index Version* and *Instrument Index Constituent* attribute classes allow representation of Index definitions and versions of these definitions such as Equity or Bond Indices.

The *Instrument Index Version* records a unique set of parameters and constituents applicable for a defined period. If the constituents or parameters of an index are changed, a new version of the index is created.

See the appendix for the full structural table description.

2.2.1.9 Table: Instrument Index Constituent

The *Instrument Index Constituent* is used to record the financial instruments that are used for an index. Details on how to set up custom indices are given in Chapter 9.

See the appendix for the full structural table description.

2.2.1.10 Table: Instrument Issuance

The *Instrument Issuance* class of attributes represents the history of Issuance events for an instrument. We do not enter into the intricacies of various offer types, issuance statuses, and issuance events. We only record issuance data for instruments that actually have been issued and record the factual issuance data.

See the appendix for the full structural table description.

2.2.1.11 Table: Instrument Issuance Date

The *Instrument Issuance Date* class of attributes represents the set of dates applicable to the history of issuance transactions recorded for an instrument.

See the appendix for the full structural table description.

2.2.1.12 Table: Instrument Rating

The *Instrument Rating* class of attributes may be used to represent the various proprietary and public ratings that are assigned to instruments and reviewed from time to time. This includes credit ratings for bonds as well as more general risk ratings.

See the appendix for the full structural table description.

2.2.1.13 Table: Instrument Relationship Condition

The *Instrument Relationship Condition* class represents constraints in terms of relationships between the financial instrument concerned and other financial instruments. Examples are the linkage of benchmarks to portfolios, programmes of short term debt issuances and derivative contract specifications from which actual contracts are derived. Actual Instruments derived from these should then follow the patterns defined in this way. See the appendix for the full structural table description.

2.2.2 The Portfolio Model

The modelling of portfolios of instruments involves the following attribute classes:

Instrument Core: The basic definition of a portfolio is set up by a table entry in the *Instrument Core* class with the attribute *Primary Asset Class* being set to 'P' (Portfolio). Different versions of portfolio definitions are specified via the *Portfolio Version* class of attributes. Links to individual instruments that form together the portfolio version are stored via the *Portfolio Constituent* class.

Performance and valuation relevant information is linked to the combination portfolio definition (instrument), portfolio version and constituent instrument using the *Portfolio Position* class.

The specification of portfolio benchmarks is done via the *Benchmark Component* class and their linkage to individual portfolios (instruments) via the *Instrument Relationship Condition* class above.

2.2.2.1 Table: Portfolio Version

The *Portfolio Version* records a unique set of parameters and constituents applicable for a defined period. Portfolio versions may be used if there are changing parameters or if constituent sets are to be frozen. See the appendix for the full structural table description.

2.2.2.2 Table: Portfolio Constituent

The *Portfolio Constituent* class of attributes is used to record the financial instruments that compose a portfolio. See the appendix for the full structural table description.

2.2.2.3 Table: Portfolio Position

The *Portfolio Position* class of attributes is used to record valuations, risk and return measures for financial instruments that compose a portfolio. See Chapter 9 for details on how to model portfolios. See the appendix for the full structural table description.

2.2.2.4 Table: Benchmark Component

The *Benchmark Component* class of attributes is used to store information on portfolio benchmarks that is to be mapped to portfolios via the Instrument Relationship Condition Class. See Chapter 9 for details on how to model portfolios. See the appendix for the full structural table description.

2.2.3 The Party Model

The Party Model is centred around the *Party Core* class of attributes. Linked to this class are classes giving details on classifications, analytical and financial information on the party. Figure 2.5 shows the conceptual structure of the submodel centred around the *Party Core* class, with the main topical areas being party description (name, etc.), contact information, accounting information, analytical information, qualitative information such as classifications or credit ratings, fund information, and relationships to other parties.

Again, we by far do not work out all the attribute classes implied by Figure 2.5. Figure 2.6 gives an overview of the classes of the party model that we use in this book, and are presented in the following sections.

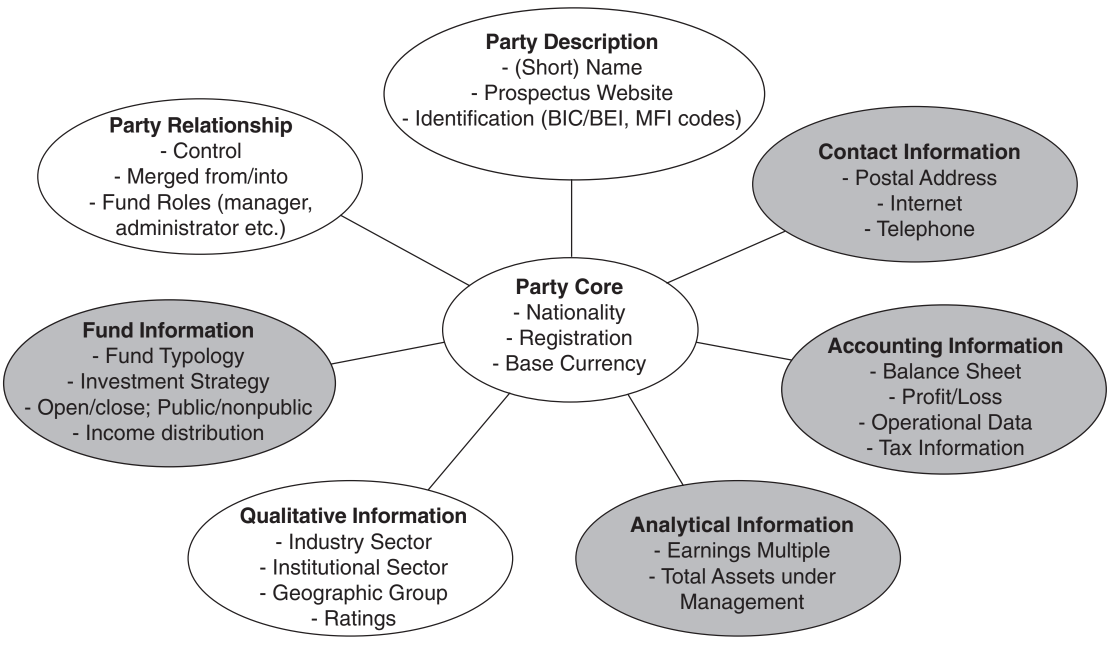


Figure 2.5: Logical structure of the full party model.

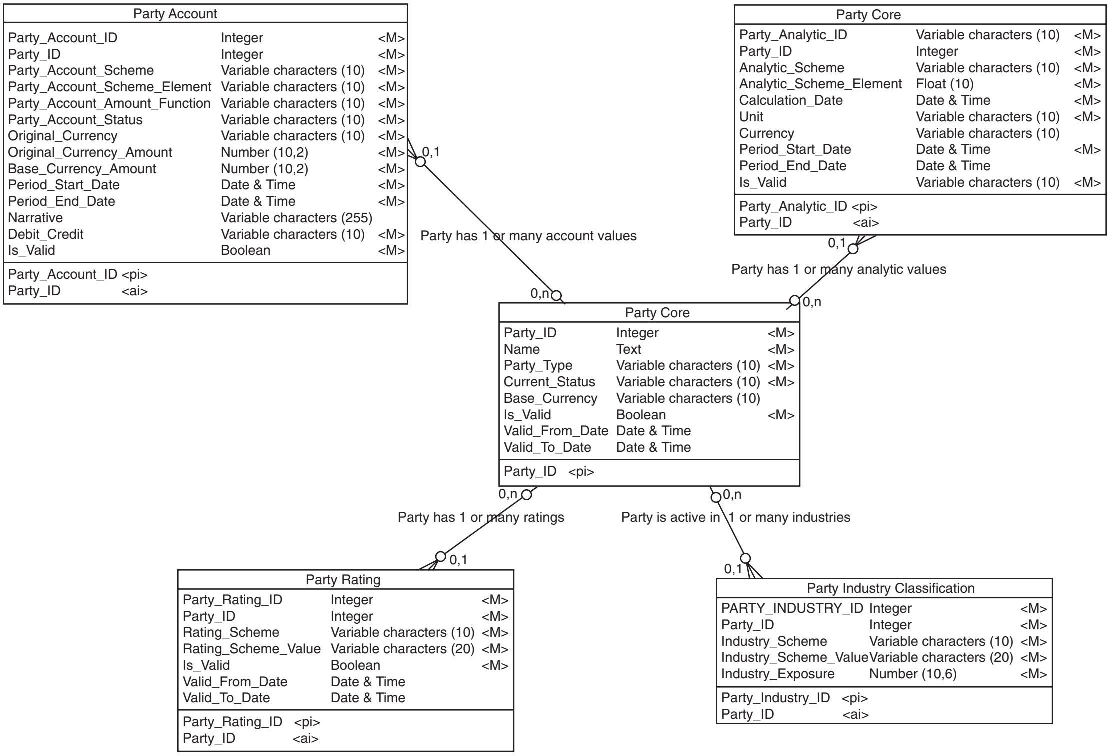


Figure 2.6: Conceptual view of the party data model classes.

2.2.3.1 Table: Party Core

The *Party Core* class stores information about organisations. It refers to organisations participating in operations with financial instruments acting in one or more of the following roles: issuer, issuer parent organisation, guarantor, depository, exchange, data source, and so on.

See the appendix for the full structural table description.

2.2.3.2 Table: Party Account

The package Party Account represents financial profiles and reports for parties. This includes income, balance sheet, and flow of funds statements, as well as lower level analytical accounts.

The *Party Account* class stores financial profiles and reports for parties. This includes income, balance sheet, and flow of funds statements, as well as lower level analytical accounts.

See the appendix for the full structural table description.

2.2.3.3 Table: Party Analytic

The Party Analytic package contains the class of attributes needed to represent imported or calculated analytic values for a party, such as earnings per share and others.

The *Party Analytic* class stores imported, derived, or calculated analytic values for a party. It stores information that can represent simple scalar analytics, such as volatility, as well as information for curve-based analytics. Curve-based analytics include, for example, default curves in the case of funds or accounting, performance and financial ratios for other issuers.

See the appendix for the full structural table description.

2.2.3.4 Table: Party Industry Classification

The *Party Industry Classification* stores information about the party's economic activity.

See the appendix for the full structural table description.

2.2.3.5 Table: Party Rating

The *Party Rating* class represents the various proprietary and public ratings assigned to parties over time. This includes credit ratings for parties as well as more generic risk and performance ratings.

See the appendix for the full structural table description.

2.2.4 The Role Model

The Role Model links the Party and Instrument Models. Parties may be represented taking on two types of roles:

- Functional, or sui generic roles, representing roles for parties depending on their function. Such roles for parties can be price or data source, exchange, clearinghouse, and funds.
- Party-instrument Roles. Examples are issuer, guarantor, originator, underwriter, among others.

Figure 2.7 shows the conceptual structure of the Role Model and its links to the Party and Instrument Models. Figure 2.8 gives an overview of the classes of the role model.

In this framework, a party can be represented as taking playing multiple roles in the lifecycle of an instrument, some or each of which is being performed in multiple functions; i.e., sui generic roles. There is a long list of potential roles parties may play. We will focus on the ones relevant for this book.

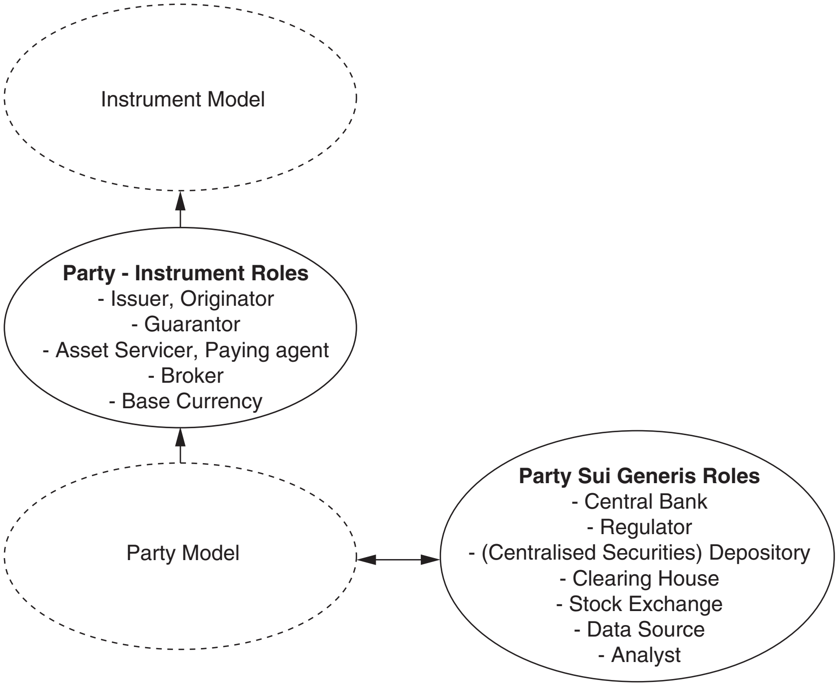


Figure 2.7: Logical structure of the full role model.

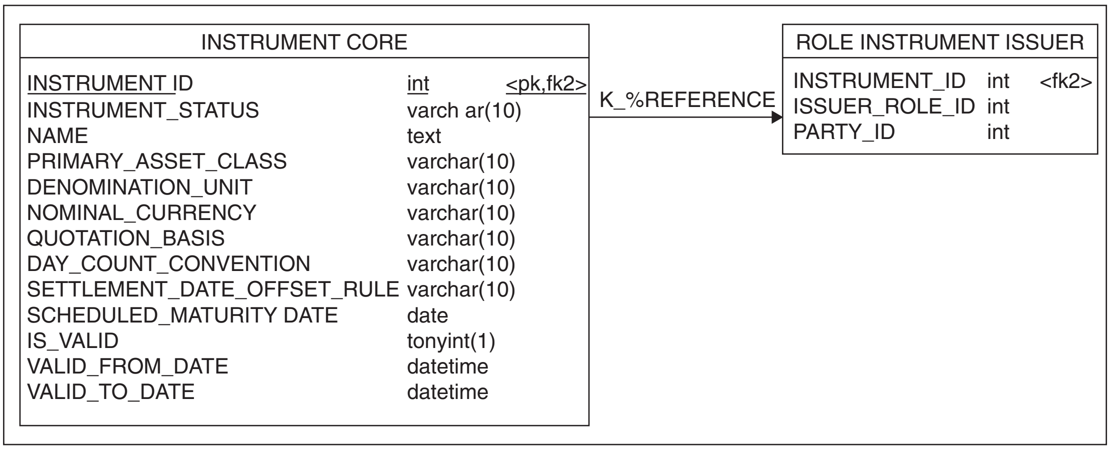


Figure 2.8: Conceptual view of the role data model classes.

2.2.4.1 Table: Role Instrument Issuer

The *Role Instrument Issuer* is used to represent the role of a Party as Issuer for an instrument. An Issuer is a party who creates an instrument and at least in the first instance is responsible and liable for any agreed payoff from an instrument throughout its life.

See the appendix for the full structural table description.

2.2.5 The Market Model

The Market Model consists of two attribute classes, *Instrument Price* and *Instrument Listing*, directly linked to the *Instrument Core* class of the Instrument Model. It allows the representation of any market or estimated prices with a comprehensive set of descriptive attributes to distinguish different types of prices. Figure 2.9 shows the conceptual structure of the Market Model and its links to the Party and Instrument Models. Figure 2.10 gives an overview of the Market model classes that are relevant for this book.

2.2.5.1 Table: Instrument Price

The *Instrument Price* stores the prices for financial instruments. This includes instrument prices in the narrow conventional sense as unit asset prices, as used with equities or percentage asset prices for instance. Also commonly used for debt instruments and in

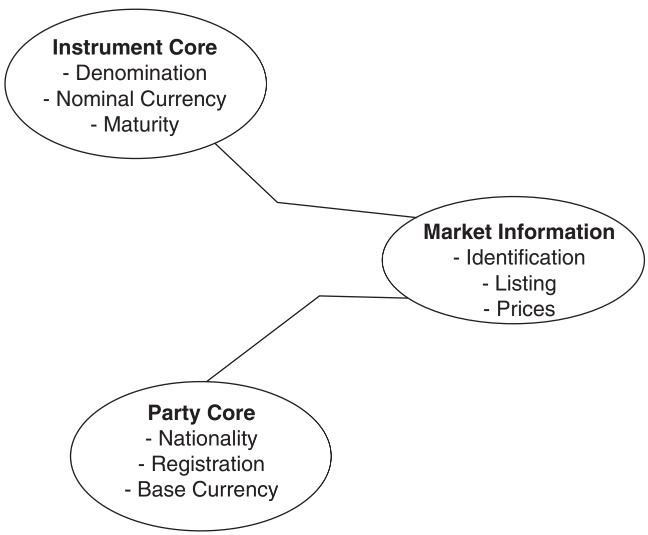


Figure 2.9: Logical structure of the market model.

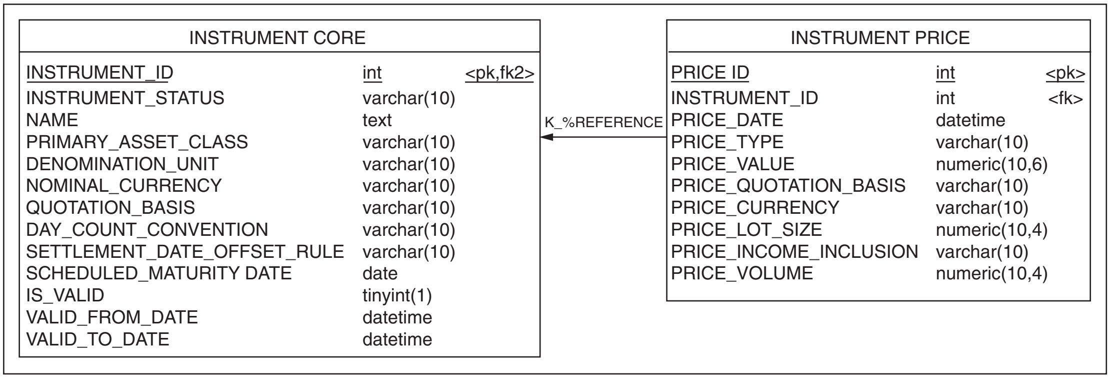


Figure 2.10: Instrument price and market data.

addition, prices in a wider sense such as index values, prices expressed as yields, and others.

See the appendix for the full structural table description.

2.3 Data Quality Management

In this book, we assume having all data in the quality and completeness required to perform the calculations of the subsequent chapters. In reality, this often will not be the case, as there is a notorious lack of data modelling standards, and the publication of financial data is scattered and, not rarely, sparse. This problem may be partially offset by a number of measures, the most proven ones of which are:

- **Multisource Databases**: Do not rely on a single data source (Commercial Data Provider or in-house data source), but combine data from multiple sources on the same real world entity. The advantage is that, with the right technology in place, you will quickly get a decent overview of the relative quality of data provided by the various sources. The disadvantage is that multisource database technology often is a nonnegligible investment.
- **Trading Policy**: Only allow investments in instruments for which you are 100 percent certain of the information available. Although this may be a reasonable strategy from a general perspective, you might not be politically in the position to enforce such a rule.
- **Ownership**: Everyone wishing to trade is responsible for the data quality of the instrument at time of trading. This has some drawbacks, if there are risk-relevant, but time-varying components. Who ensures quality here?

- **Store information** as **close to source format** as possible. Every friction or transformation of data from the original source (usually the issuance prospectus or balance sheet publication) induces the risk of errors in the data by mistyping, mapping, or processing errors.

In this chapter, we defined a robust data model that will be able to support all calculations outlined in this book. You should keep in mind however that any data model and the results of any subsequent processing and calculation is only as good as the data being fed to it.

CHAPTER 3

Modelling Financial Instruments

After a short detour on the nature of data sources for financial instruments, we model a real-life example of a straight fixed coupon bond. For most information items, this example is illustrative and variations can easily be derived from looking at the appropriate code list domains. A particular challenge is the modelling of cash flow structures, one of the most important, but also most complicated aspects of financial instruments. We therefore present the principles of cash flow modelling and discuss the modelling of cash flows for an illustrative set of instrument types.

3.1 Modelling Financial Instruments

With the data model at hand, we can now use it to represent information on real financial instruments. To illustrate this we start with modelling a **straight fixed coupon bond**. This is a debt instrument with no embedded options (= straight) and no variation in the income promised to the investor (= fixed coupon).

After a short detour on the nature of data sources for financial instruments (Section 3.1.1), we model a real-life example of such a bond in Section 3.1.2. For most information items, that example is illustrative and variations can easily be derived from looking at the appropriate code list domains. You will also have the opportunity to sharpen your intuition later in this book when we use the model to extract the information required for the various calculation algorithms.

A particular challenge is the modelling of cash flow structures, one of the most important, but also most complicated aspects of financial instruments. We will therefore present the principles of cash flow modelling separately and discuss the modelling of cash flows for an illustrative set of instrument types in Section 3.1.3.

Finally, we take a look at the modelling of market information; that is, instrument listings and real prices.

3.1.1 Where Does the Information Come From?

Information on financial instruments is provided by the issuer to the investor(s) and, if the instrument is traded on regulated markets, to the general public. The structure of the instrument is communicated via a so-called issuance prospectus, a comprehensive legal documentation of the financial contract represented by the instrument. This usually is complemented by the regular reports on the business activities of the issuing organisation.

However, as with other legal contracts, contracting parties often have an interest not to disclose too much information to any third party. So, when there is no legal obligation to publish you may expect to find only a minimum of information available, in particular if the instruments are bilateral in nature (also often called Over-the-Counter or OTC instruments), and hence the conditions are not accessible by the general public. This illustrates the contractual nature of financial instruments: They are *a priori* bilateral contracts where there is no need to give insight to contractual details except to the parties concerned. It may sound strange, but compared to the bulk of instruments issued, the widely known securities are rather the exception than the rule.

On the other hand, if you are not in the business of constructing financial statistics or are not one of the parties concerned, it should not bother you too much that you cannot get sufficient information on (OTC) instruments.

For publicly traded or tradable instruments, most industrialised countries have established legal obligations on the issuers to publish comprehensive information on a regular basis, not only on the structure and status of the instrument, but also on their ongoing business.[1] These requirements are the stricter, the more the instrument is accessible by the general public. They are particularly strict for securities, instruments traded on regulated markets such as stock exchanges or multilateral trading platforms, as there is a general interest to protect nonprofessional investors.

So far the theory. In practice, even with widely traded securities (notably debt instruments), there exists a significant amount of opacity. Information is accessible, but not in straightforward and certainly not standardised ways. This starts hurting when you need to collect and process high quality information on a large number of financial instruments. And it is here where Commercial Data Providers (CDPs) come into play: These companies specialise in maintaining good relationships with issuers and regulated market organisations,

---

[1] In the European Union, these obligations are based mostly on the European Union Prospectus Directive (2001/34/EC) and the European Union Transparency Directive (2004/109/EC). The first specifies what needs to be published, and the latter specifies when and in what format it needs to be published. The directives are accompanied by guidelines giving more details to the various requirements. As European Legislation is implemented at the country state level, the exact national obligations may vary from these rather general legal prescriptions.

collect a maximum of reference (static) and price (dynamic) information on financial instruments, thus making life a lot easier for financial market participants.

3.1.2 A Real-Life Example

We begin with the information shown in Figure 3.1, a so-called "term sheet," summarising key information about a financial instrument; that is, its "terms and conditions" (hence the

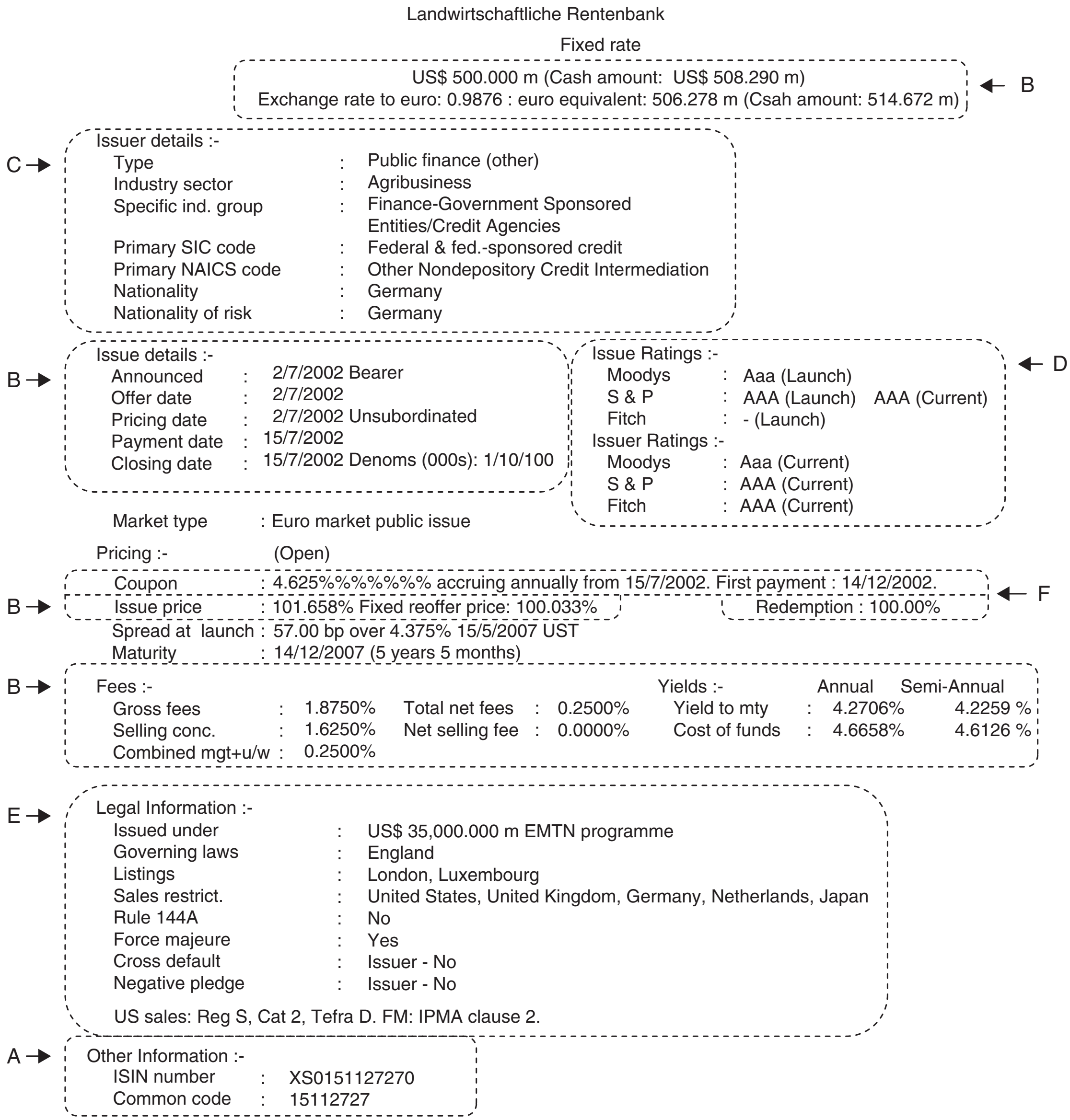


Figure 3.1: Term sheet of a Straight Coupon Bond as of 16 April 2006.

name). If you subscribe to a Commercial Data Provider, you will have access to much more information. But for our purpose of clarifying the data modelling principles, this is even more than enough.

- Any term sheet implicitly defines the conventions and measures that determine the instrument throughout its lifecycle. This “core information” about a financial instrument does not and cannot change, as it would fundamentally alter the contractual relationship between issuer and investor. We will discuss its identification and storage in Section 3.1.2.1. Beyond that, we can distinguish several broad categories of information: **Instrument identification**: Most financial contracts are registered with national or supra-national authorities, even if their exact content is not going to be made public. The international standard for instrument identification is the International Securities Identification Numbering system (ISIN), defined as an ISO standard (ISO 6166), and incorporates most national securities identifiers (such as the CUSIP from the United States, the SIRENE from France, or the WKN from Germany). We show how to store instrument identifiers in Section 3.1.2.2.
- **Issuance information**: The selling phase of the instrument. Very short with OTC instruments, it often is quite lengthy, varied, and rather complicated with publicly traded instruments. A significant amount of data generated in this process, such as nominal amount issued, issue price, or issuance dates, is key for the further metrics to be calculated for the instrument. We discuss that data in Section 3.1.2.3.
- **Information about the issuer** of the instrument, notably the country of legal incorporation (residence, nationality) and sectoral classification: As in any contractual relationship, knowing your counterparty is always of help. We treat the modelling of issuer-related information in Section 3.1.2.4.
- **Rating information** evaluating the creditworthiness of the instrument and the issuer: For investors, this is essential information on the probability that the cash flows promised by the issuer are actually going to be paid out. The modelling of rating information is shown in Section 3.1.2.5.
- **Miscellaneous legal information**, such as the governing law, sales restrictions and whether it can be traded in the United States (Rule 144A, etc.): Modelling of these is usually a mere classification exercise, so we will not enter into its details.
- **Cash flow information**: We discuss this key instrument characteristic in profound depth in Section 3.1.3. There we will start with the simple example of the present term sheet, representing a straight fixed coupon bond, and then will explore further, more complicated structures and their modelling.

3.1.2.1 Instrument Core

The ever-equal starting point for storing information on a financial instrument is the *Instrument Core* class of attributes, where we initialise the instrument and put some basic information as reference point for all other classes. Unfortunately, our term sheet is not too illustrative on the measurement units and general conventions applicable to the instrument. However, based on common market conventions, we can infer much of the missing information[2] and fill in the attribute class as shown in Table 3.1. Details on the code list specific values may be found in the Appendices.

As you can see, the *Instrument Core* class indeed sets the scene in terms of conventions and definitions applicable to the full life of the financial instruments, and has an influence on many analytical calculations. These definitions cannot be changed without fundamentally altering its character—which would make the contractual basis between issuer and investor obsolete.

3.1.2.2 Instrument Identifiers

Similar to the information stored in *Instrument Core*, the identification via official instrument identifiers is fixed and unique during the life of the instrument.

> Note that official security identifiers, like the ISIN code, can be reused some time after the maturity/expiry of an instrument. For this reason you should not make the unique identification of an instrument in your database reliant on official security identifiers.

However, as there is no rule on how many and what type of identifiers an instrument can have, we model this information using a repetitive structure (table) allowing multiple entries per instrument.

As most instruments these days are required to have an International Securities Identification Number (ISIN) code, it is not a bad idea for you to look out for the ISIN code as standard identifier of any traded security. Many Data Source specific codes in fact are listing and not instrument specific; that is, an instrument listed in multiple markets (exchanges) will have multiple identifiers, thus making unique instrument identification difficult. The reason lies in the market price orientation of many commercial data providers, for whom the identification of a specific listing is more important than that of a unique instrument. See Table 3.2 for an example of how to capture identifier information using the Instrument Identifier class of attributes.

---

[2] In practice, though, you will usually get this information from your Commercial Data Source.

Table 3.1: Real example data mapping to the *Instrument Core* class of attributes

Attribute Name	Value	Comment
Instrument ID	000001	Unique identifier of the instrument, automatically allocated by the system. In our example we take this to be 000001.
Instrument Status	ISS	Issuance status of the instrument. It is obviously successfully "Issued".
Name	Rentenbank 4.625% 14/12/2007	Usually a composite of key information on the instrument. Here we use a standard for debt instruments: ⟨name of issuer⟩_⟨annual coupon rate⟩_⟨maturity date⟩.
Primary Asset Class	D	Debt instrument.
Denomination Unit	CCY	Quantity in which the instrument itself is measured. We assume that this is the minimum monetary amount in Nominal Currency; i.e., 1 US Dollar cent.
Min Denomination Amount	0.01	No information given. We assume that the investor can hold the minimum monetary amount possible.
Nominal Currency	USD	The instrument is issued in US Dollars; i.e., unless specified otherwise, all monetary amounts are measured in US Dollars.
Quotation Basis	PCT	The instrument is quoted in percent. This information is derived form the issue price being quoted in percent. As compared to the [Denomination Unit] attribute, this attribute specifies in which denomination a price of the instrument is quoted (defined) in the markets.
Day Count Convention	ACT_ACT	No information given on how year fractions are to be calculated. We assume the most accurate convention, taking the actual number of calendar days between two dates and dividing it by the actual number of days in the current year.
Settlement Date Offset Rule	T+1B	No information given. We use a market standard convention: Settlement takes place 1 business day after Trading date.
Scheduled Maturity Date	14.12.2007	Maturity date: Date on which the instrument is scheduled to expire; i.e., the financial contract ceases to be of legal relevance.
Is Valid	Yes	The record represents a valid, real life instrument.
Valid From Date	02.07.2007	We assume that the instrument is a valid record as of the first date on which information is given to the markets: Announcement date.
Valid To Date	NULL	A priori, the validity of the instrument is unlimited.

Table 3.2: Instrument identifiers stored using the Instrument Identifiers class

Attribute Name	Value 1	Value 2	Comment
Instrument Identifier ID	(Auto)	(Auto)	Automatically generated by the system.
Instrument ID	000015	000015	Internal system ID of the instrument.
Numbering Scheme	ISIN	COMM	We observe two identifiers, one ISIN code and one Common code.
Identifier Value	XS0151127270	15112727	

3.1.2.3 Issuance Information

The issuance process is the initiation moment of the instrument, where it is being offered (tendered) and sold to one or many investors. Next to the cash flow structure, the issuance process and corresponding modelling is one of the most challenging areas of financial data modelling, as there are multiple ways of issuing financial instruments, each generating a wide range of information.

Although we acknowledge that for some applications having very detailed issuance information is of key interest, it is usually enough to capture a relatively small set of key data. We will hence focus in this book on the most needed concepts and their modelling:

- **Issuance transaction and type**: There are multiple ways to issue an instrument, each representing a different issuance transaction type. The corresponding code list in the Appendices gives a flavour of the wide variety of different possibilities.
- **Issuance dates**: In many applications people like to speak of *the* issue date of an instrument. However, in practice, the different issuance transaction types each imply a series of dates all related to the issuance process. What is now to be used as *the* issue date depends on the data you have available (sometimes nicely restricting your choice) and the application you have in mind. In this book, we model the full set of possible dates (see the Appendices for the code list) in a dedicated attribute class.
- **Amount issued**: Although not particularly relevant for pricing or performance measurement, the amount issued (or, if related, the amount outstanding) of an instrument is a very intuitive measure of the "volume" of the instrument. It measures the debt load the issuer added to his or her balance sheet by issuing the instrument, but also can give a strategic indication on the market size for the instrument, and hence the degree of competition in the market. In the event of a credit crunch or merger/squeeze out, having exact information on the amount issued can be crucial to both regulators and investors.
- **Issuance tranche**: After the initial public offering of an instrument, nothing prevents the issuer from increasing the volume of that instrument by launching subsequent

issuance tranches. For example, assume that at an initial public offering the issuer launched a debt instrument with an amount issued of EUR 1 billion. One year later, the issuer wishes to take on more debt under the same conditions as before. So he or she launches a Secondary Public offering of a second tranche with an amount issued of another EUR 500 million, making the total amount outstanding after that transaction to be EUR 1.5 bn. The same applies to equity shares (stocks), where it is not uncommon that companies raise their capital by issuing new shares. In the case of open funds, where units are constantly issued and repurchased, the tranche does not make too much sense. We capture this feature by modelling issuance transactions in a repetitive structure, with each transaction record having a tranche number allocated to keep track of multiple tranches.

- **Issue price**: Measures what has been paid per nominal value (debt), share (equity), or unit (funds) on issuance by the investor(s). The issue price can be paid in cash or (potentially) in kind (i.e., other instruments). It is transaction specific and hence stored in the *Instrument Issuance* class.
- **Accrued coupon at issue**: Sometimes interest bearing instruments have an initial gift to the investors (which they have to pay for, of course), the coupon already accrued at issue. The reasons for offering this may vary, ranging from tax avoidance to making the instrument more attractive in a competitive environment, while not wanting to raise too much the regular interest to be paid. This information is also transaction specific and stored in the *Instrument Issuance* class.
- **Issuance fees**: We do not model here the fees charged by the issuance syndicate, helping the issuer getting the instrument out and sold in the market. This information is not needed for pricing, risk, and performance measurement for the instrument.

With these concepts in mind, we can now proceed to storing the relevant issuance information in our real-life term sheet. The modelling of issuance information works around two classes:

- The *Instrument Issuance* class captures all information that is specific to a single issuance transaction (see Table 3.3).
- Linked to each entry of the issuance transaction class, representing a specific issuance are one to many entries in the *Instrument Issuance Date* class, capturing an *a priori* unlimited number of issuance transaction related dates (see Table 3.4).

The example nicely provides us with five issuance dates, each representing a different milestone in the regular issuance process, captured by the record in the Instrument Issuance table.

Table 3.3: Real example issuance data

Attribute Name	Value	Comment
Instrument Issuance ID	000001	Unique identifier automatically allocated by the system. In our example we assume this is 000001.
Instrument ID	000015	Internal system ID of the instrument.
Issuance Transaction Type	IPO	No information provided. We assume that it is an Initial Public Offering.
Number Of Tranches	1	No information provided on whether this is a multitranche issue. For the time being we assume that there is only one single tranche, represented by the instance of this table.
Name Of Tranche	NULL	No tranche-specific name provided.
Issuance Paid Amount	500,000,000.00	Nominal debt of the instrument. Note that in our example the issuer managed to receive more than that in cash.
Issuance Amount Currency	USD	US Dollar
Issuance Paid Number	NULL	Only applicable of instrument is quoted in units.
Issue Price	101.658	Issue price in percent.
Consideration Currency	USD	US Dollar
Consideration Cash	101.658	
Consideration Physical	NULL	All is paid in cash.
Consideration Instrument ID	NULL	No information available.
Accrued Coupon At Issue	NULL	No information given.

3.1.2.4 Issuer Details

On the issuer of the instrument, Figure 3.1 tells us only some key information, notably what type of institution it represents. We see several classifications, all saying the same thing: The issuer is a German federal government-sponsored credit institution that operates in the agribusiness sector.

We model the issuer by generating a new record in the party-related part of the data model and then linking that record to the instrument record via the *Role Instrument Issuer* class.

We start with a new entry in the Party Core class as in Table 3.5.

Table 3.4: Real example set of issuance dates

Attribute Name	Value 1	Value 2	Value 3	Value 4	Value 5
Issuance Date ID	000001	000002	000003	000004	000005
Instrument ID	000015	000015	000015	000015	000015
Instrument Issuance ID	000001	000001	000001	000001	000001
Issuance Date Type	ANNOUNCEMENT	OFFER	PRICING	OFFERCLOSING	PAYMENT
Issuance Date	02.07.2002	02.07.2002	02.07.2002	15.07.2005	15.07.2002

Table 3.5: Real example issuer party record in the *Party Core* class

Attribute Name	Value	Comment
Party ID	000020	Unique identifier of the party representing the instrument issuer. The number is allocated by the system automatically. In our example we take this to be 000020.
Name	Landwirtschaftliche Rentenbank	Name of the issuer party.
Party Type	COM	Company or other commercial legal entity.
Current Status	ACT	Active.
Domicile Country	DE	The party is legally incorporated and registered in Germany.
Registration Place	NULL	No information given.
Base Currency	EUR	The business of the party is transacted and reported on in Euro.
Is Valid	YES	The record represents a valid, real-life instrument.
Valid From Date	NULL	No information is given as to when the issuer party was incorporated.
Valid To Date	NULL	The party is still active, hence no suspension date is available.

As for the issuer classification we use an instance of the *Party Industry Classification* class, specifying the International Standard Industrial Classification where we find the appropriate scheme value on the http://unstats.un.org/unsd/cr/registry/regcst.asp?Cl=27 web site (ISIC Rev. 4) to be K.641: "Financial and insurance activities—Monetary intermediation." We assume that this is the only industry where the issuer is active, so we set the industry exposure to 1.0. In our data model we store an instance of the class as shown in Table 3.6.

Table 3.6: Industry classification of the issuer party via the *Party Industry Classification* class

Attribute Name	Value	Comment
Party Industry ID	(Auto)	Automatically generated by the system.
Party ID	000020	System ID of the issuer party.
Industry Scheme	ISIC	International Standard Industrial Classification.
Industry Scheme Value	K.641	Specifies the specific sector classification (element) for a given sector classification scheme to which the party being classified is exposed or belongs.
Industry Exposure	1.0	Specifies the extent to which the Party is exposed to a specific classification.

Table 3.7: Issuer—instrument linkage via the *Role Instrument Issuer* class

Attribute Name	Value	Comment
Issuer Role ID	(Auto)	Automatically generated by the system.
Instrument ID	000015	Internal system ID of the instrument.
Party ID	000020	Internal system ID of the party.

Issue Ratings:-			
Moodys	-	Aaa (Launch)	
S & P	-	AAA (Launch)	AAA (Current)
Fitch	-	- (Launch)	
Issuer Ratings:-	-		
Moodys	-	Aaa (Current)	
S & P	-	AAA (Current)	
Fitch	-	AAA (Current)	

Figure 3.2: Rating information of the Term Sheet example.

Finally, we create the link between the instrument with INSTRUMENT ID = 000015 to the issuer party with PARTY ID = 000020 via an instance of the *Role Instrument Issuer* class as shown in Table 3.7.

3.1.2.5 Ratings

We have quite some information on ratings, one set for an instrument specific rating and one for issuer party-specific ratings as recaptured in Figure 3.2.

Distinction is further made between a rating "at launch," supposedly at the time of issuance, and a "current" rating. For the time of issuance we take it to be the earliest issuance-related date 2 July 2002. We take the current date from Figure 3.1 to be 16 April 2006, the date of term sheet generation. We assume that if there is no difference between ratings at launch and those labelled current, that there has been no movement in the ratings between the two dates. All ratings are further taken to be the respective long-term ratings (see the Appendices for a list of possible rating types).

These considerations allow us to generate two instances of instrument-specific ratings, one from Moody's and one from Standard & Poor's, as shown in Table 3.8.

In addition, we can specify three instances of issuer-specific ratings that are stored in the *Party Rating* class, linked to the issuer Party ID as in Table 3.9. The only difference to Table 3.8 is that we have no indication as of when the respective ratings are valid for the issuer organisation.

Table 3.8: Instrument related ratings in the *Instrument Rating* class of attributes

Attribute Name	Value 1	Value 2	Comment
Instrument Rating ID	(Auto)	(Auto)	Automatically generated by the system.
Instrument ID	000015	000015	Internal system ID of the instrument.
Rating Scheme	MDYLTCO	SNPLTIR	Respective long-term issue rating.
Rating Scheme Value	Aaa		Value from term sheet.
Is Valid	YES	YES	
Valid From Date	02.07.2002	02.07.2002	See earlier assumptions.
Valid To Date	NULL	NULL	No information available.

Table 3.9: Issuer related ratings in the *Party Rating* class of attributes

Attribute Name	Value 1	Value 2	Comment
Party Rating ID	(Auto)	(Auto)	Automatically generated by the system.
Party ID	000020	000020	Internal system ID of the Issuer Party.
Rating Scheme			Respective long-term issuer rating.
Rating Scheme Value			Value from term sheet.
Is Valid	YES	YES	
Valid From Date	NULL	NULL	See earlier assumptions.
Valid To Date	NULL	NULL	No information available.

Legal Information:-		
Issued under	:	US$ 35,000.000 m EMTN programme
Governing laws	:	England
Listings	:	London, Luxembourg
Sales restrict	:	United States, United Kingdom, Germany, Netherlands, Japan
Rule 144A	:	No
Force majeure	:	Yes
Cross default	:	Issuer - No
Negative pledge	:	Issuer - No

US sales: Reg S, Cat 2, Tefra D, FM: IPMA clause 2.

Figure 3.3: Miscellaneous information of the term sheet example.

3.1.2.6 Miscellaneous Information

There is a number of additional information on the term sheet, shown in Figure 3.3, which is mostly of a legalistic nature.

Except for the market listing information, their modelling involves a classification-style addition of data, as illustrated in the section for the Party Industry classification. An appropriate attribute class could be labelled something like *Instrument Conformance*. Because they are not relevant for the purposes of this book we will not enter this subject in more detail.

3.1.3 Modelling of Cash Flow Structures

Equipped with the modelling approach, we now have a closer look at one of the most tricky aspects of financial instruments: the cash flow structure.

Section 3.1.3.1 gives a short phenomenology of cash flow structures and describes their logic and decomposition options. After a brief discussion of the pros and cons we present two modelling options:

- Precalculated storage in Section 3.1.3.5.
- Storing the structural elements of cash flow structures, and the logic of this approach is explained in detail in Section 3.1.3.6. A range of illustrative examples is given in Section 3.1.4.

3.1.3.1 What Is a Cash Flow Structure?

A cash flow is a monetary payment between two parties of a contractual agreement. A **cash flow structure** is the (possibly predefined) series of cash flows of a financial instrument. The cash flow structure of any financial instrument, however complicated, can

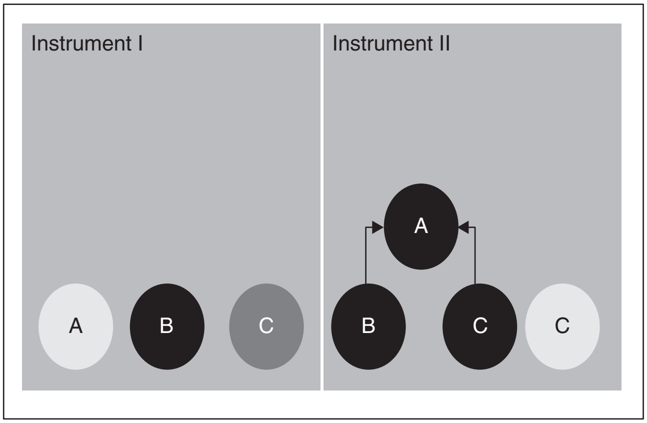


Figure 3.4: Combination of elementary cash flow structures to set up more complicated cash flow structures.

be decomposed into a small set of elementary cash flow structures, so-called **atomic cash flow elements**.

In Figure 3.4 this principle is illustrated for the case where a cash flow structure is represented by a basket of atomic cash flow elements (Instrument I) A, B, and C. They could represent, for example, a single, bullet-style, fixed redemption cash flow element, a fixed coupon cash flow element, and an option element that allows the issuer to call (or “redeem”) the instrument early at a fixed price.

Instrument II in the same figure is an example of where the basket of atomic cash flow elements used to represent the instrument contains a structure of atomic elements. This could be for instance an instrument where the coupon of which is driven by the difference of the value or price of two independently moving underlying instruments with a cap or floor on the resulting coupon. From the point of view of the investor(s), four cash flow types can be distinguished:

- **Principal payment(s)**: Initial cash flow(s) from the investor(s) to the issuer. This is the “money invested” in the financial instrument. The payout to the issuer may be in a one-off or staggered, stepped mode. Down payment of an investor to the original issuer or current holder or OTC counterpart of the financial instrument at time of issuer or purchase. It signals the entering of the investor into the investment-relationship with the issuer of the financial instrument and originates the legal claim for future cash flows from this investment. This down payment may be done in multiple steps, called **tranches**.
- **Income payment(s)**: Intermediary payments from the issuer to the investor(s). Income payments may be specified fully in advance (fixed or stepped), be depending

on a reference rate (floating or index linked), or be left fully at the discretion of the issuer (usually in exchange for ownership rights in the investment represented by the financial instrument). Payments from the issuer to the (current) holder of the financial instrument in recompense of him or her bearing the financial market, credit risk associated with the investment. With debt instruments, income payments come in as coupon payments, which may follow a predefined or an irregular schedule in timing or value. Income on equity instruments usually are paid in the form of dividends, being irregular both in timing and value by their very nature.

- **Redemption payment(s)**: Repayment of the initial down payment from the issuer to the (current) holder/investor. As with the initial down payment, the redemption payment may be done in multiple, possibly irregular steps. Repayment of principal to the investor(s) is by the issuer to the investor. Repayments may be done in one go (bullet redemption), in a predefined sequence (stepped), conditional (index- or event-linked), or as a function of some underlying business (irregular; e.g., depending on the repayments of underlying mortgages or loans in the case of Asset Backed Securities).
- **Options, representing special rights on one of the previous cash flow types**. These rights refer mostly to the timing and modalities of the redemption payment(s). We give examples for optional cash flows in Sections 3.1.4.5 and 3.1.4.10.

Any cash-flow structure may be represented graphically by a bar chart with a horizontal axis that measures the times between consecutive payments, and a vertical axis that measures the amounts of cash flows. The cash flows take positive values when they are received (from the buyer's perspective) and are negative otherwise. Figure 3.5 illustrates this graphical presentation technique using the example of a straight (no embedded options) fixed coupon bond with an initial purchasing price (negatively defined),

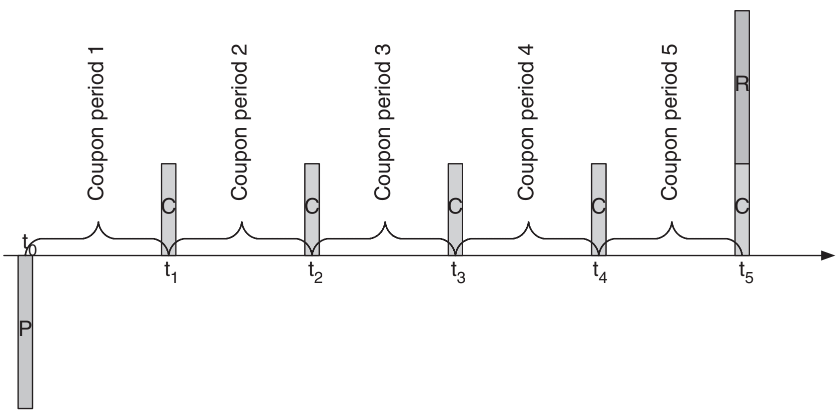


Figure 3.5: Cash flow of a straight fixed coupon debt security with bullet redemption payment.

subsequent coupon payments, and a redemption payment (positively defined), the latter being possibly accompanied by a final coupon payment. The scheme is from the perspective of the buyer who wishes to value the security, which he or she (potentially) intends to buy.

To illustrate these concepts, Figure 3.5 shows the predefined cash flow structure of a straight fixed coupon bond[3] with bullet redemption. The bond is issued at date t_0 with an issuance payment of P; has subsequent coupon payments of a fixed rate C at dates t_1, t_2, t_3, and t_4; and has a final (combined) redemption payment of R and coupon payment of C at (maturity) date t_5. In the example, the payment date of the last coupon coincides with the redemption date of the principal from the issuer to the investor. With this payment, all claims between issuer and investor are settled.

If a specific set of cash flows are agreed upon at issuance, we speak of the instrument as having a **cash flow schedule**. It usually consists of the amounts to be paid as well as the dates on which they are promised to be paid. Two types of cash flow schedules can be specified at the time of issuance:

- The **income schedule** specifies the structure of the income payments.
- The **redemption schedule** specifies the structure of the redemption payments.

Figure 3.5 also highlights the intertemporal nature of a cash flow structure, thus the timing between the individual occurrences of the cash flows is for both risk assessment and pricing purposes. Because the income payments are usually specific in annual rates of some underlying principal or external reference rate/index, the time elapsed between two cash flows is measured in **year fractions**. Year fractions are measured as the ratio of the number of days between two business days to the overall number of days in a given year, according to a predefined calculation schedule. For the case that a predefined cash flow date falls on a nonbusiness day, predefined **date roll rules** specify which date shall be used instead of the date representing a nonbusiness date. Whether a given date is a business date or a nonbusiness date is determined by business **calendars**, which depend on the market on which the security in question is being traded.

All these rules and conventions can be specified and stored at multiple layers of the data model. We can imagine for example a specific set of rules for the income cash flow schedule, another one for the redemption schedule, and so on. In this book, we assume that a single set of rules is specified for all cash flows of the instrument, and hence the corresponding attributes may be found in the *Instrument Core* class. This, however, is a

[3] The term **straight** denotes the absence of any embedded optionality potentially affecting the coupon or redemption schedule.

mere convention to make the exposition easier. You should not refrain in your own application to specify a more flexible setup whenever needed.

3.1.3.2 Empirical Cash Flow Structures

The full combination of cash flows exchanged over the lifetime of an instrument is called the **cash flow structure**. It depends on the general nature of the instrument and the level of precision at which cash flows are being pre-agreed at issuance:

- **Debt instruments** typically contain very precise specifications on income and redemption payments, usually in the form of cash flow schedules. In exchange, the claims on the promised cash flows are the only legal liability between issuer and investor(s).
- **Equity shares** are the other extreme: By paying the principal, the investor acquires ownership rights on the (material and immaterial) assets proportional by the nominal value of the share in relation to the equity of the company issuing the shares. However, except for a sale of the shares in the secondary market or in the event of an external take-over bid there is no redemption foreseen. In addition, income payments are to a very large degree at the discretion of the management of the company, so that—when buying equity shares with no ownership agenda—the investor to a very large degree speculates on a rising price of the share in the secondary stock market.
- A special case is **preference shares**, which resemble perpetual debt instruments, but—for accounting, tax, and creditworthiness reasons—are treated as nondebt instruments.[4] They tend to imply weaker or no ownership rights in exchange for an income cash flow guarantee or higher income promise.
- **Fund shares** are located in the middle between equity shares and debt instruments.

Table 3.10 gives a broad, though not exhaustive, orientation of the mapping of instrument types to cash flow structures. Basically any day, a new such combination can be invented by a financial engineer, or a client considering the new structure to be exactly fitting his or her needs.

3.1.3.3 Modelling Philosophies

Concerning the actual cash flows (i.e., the recording of historical payments, fixings, etc.), the modelling approach is straightforward: Dedicated tables, possibly by cash flow type, do

[4] In the hierarchy of claims, in the event of a default of the issuing company, preference shares usually range among other equity, and hence have lower priority of reimbursement. Collecting money via the issuance of preference shares thus has the effect of raising the issuers' creditworthiness profile.

Table 3.10: Cash flow structures by instrument and income type

Instrument type	Income type	Principal payment(s)	Income payment(s)	Redemption payment(s)
Debt instrument	Fixed	One-off or in tranches	Income schedule	Redemption schedule
	Stepped	One-off or in tranches	Income schedule	Redemption schedule
	Perpetual	One-off or in tranches	Income schedule	None
	Zero coupon	One-off	None	Redemption schedule
	Floating	One-off or in tranches	Linked to rate	Redemption schedule
	Index-linked	One-off or in tranches	Linked to index	Redemption schedule
Equity shares	—	One-off or in tranches	Discretionary	No
	Preference shares	One-off or in tranches	Income schedule	None
Fund shares	—	One-off	Discretionary	No
	Closed	One-off or in tranches	Cash flow schedule	Redemption schedule

the trick. Principal cash flows come along only as actual payments and hence can be stored alongside other issuance-related information. Opinions are divided on how to treat scheduled cash flow structures. There are two philosophies on how to model and store such information:

A. Calculate the series of promised cash flows at issuance and store them for later use bit by bit. We illustrate this approach using the real example in Section 3.1.3.5: Calculate the exact expected cash flow dates and amounts and store them in a dedicated table.

B. Store the schedule as defined in the prospect, term sheet, or contract and calculate the matrix of cash flows and calculate the matrix whenever needed. Use the structural information about the scheduled cash flows, such as start date, end date, interval, cash flow value, and so on, and store them in a dedicated, structural model.

Approach B has three advantages over A:

1. Except for instruments with very short cash flow schedules (such as short-term debt), you tend to use less storage capacity of your system. The actual cash flow payments information can be substantial if stored individually, but often can be neatly summarised in one or two table entries.

2. You stay closer to the original data, and hence incur less risk of data corruption. Cash flow structures usually are specified in structural mode in the issuance prospectus of the instrument: If you agree on a loan with your bank you also usually do not say "I pay x amount of money on date A, x amount on date B, and so on," but rather "I pay x amount of money every second day of the month for z years."
3. Most importantly, certain cash flow structures simply cannot be calculated up front, such as index-linked cash flows, or it would be overly costly to do so. This is the case if you want to store information on different schedules, for example, a payment, a fixing, and an ex-coupon schedule.

For these reasons, we give a preference to the structural storage approach, even though it means that for pricing purposes you have to add the extra step of calculating your (current) cash flow matrix. For pricing, risk, and performance calculations it is ultimately the matrix that is relevant.

An important development principle says that you should avoid any logical frictions in the design and setup of your application. Using both methods will nearly certainly lead to unnecessary duplication of information, meaning a violation of this principle, albeit a minor one.

Still, you might or will have a data source that can provide you only with precalculated information, due to internal software design decisions, so you have no choice other than to use the information in this format. In many cases, if you wish to keep track of actual payments, you have to store both structural and actual information.

It is therefore useful to be prepared for both approaches, and to design your downstream calculations in a way that they are capable of dealing with both types of information.

3.1.3.4 A Conceptual Data Model for Cash Flow Structures

We propose a data model that allows us to capture cash flow information separately by type of cash flow in a series of tables, depending on the nature and information content of the cash flow structure:

- For **(principal) payment information** the system captures planned or actual information on:
  - The full issuance transaction (planned or actual) using the Instrument Issuance and the (linked) Instrument Issuance Date tables.
  - The current outstanding (or issued) principal amount in an Instrument Issuance Position table.
- For **income information**:
  - If a schedule exists, using the Instrument Cash Flow Element and the (linked) Instrument Cash Flow Schedule tables.

  - If, out of a schedule, a certain income amount or rate has been fixed, the fixing information is stored in the Instrument Cash Flow Fixing table.
  - If income actually has been paid out whatever the background (discretionary or scheduled/fixed) the corresponding payment information is stored in an Instrument Income Payment table.

- For **redemption information**:
  - If the redemption is prescheduled, the scheduling information is given in the Instrument Cash Flow Element and (linked) Instrument Cash Flow Schedule tables.
  - If an actual redemption payment happened, the corresponding information is stored in the Instrument Redemption table.

The linkage between cash flow related information, informational content, and data model tables is shown in Figure 3.6. When tables are used for the storage of similar information from different cash flow types, the cash flow type is identified via a "type" or "scheme" identifier in the table itself. As in theory all or most combinations between the various cash flow types and subtypes are possible, the full appreciation of a given financial instrument can be done only on a case-by-case basis.

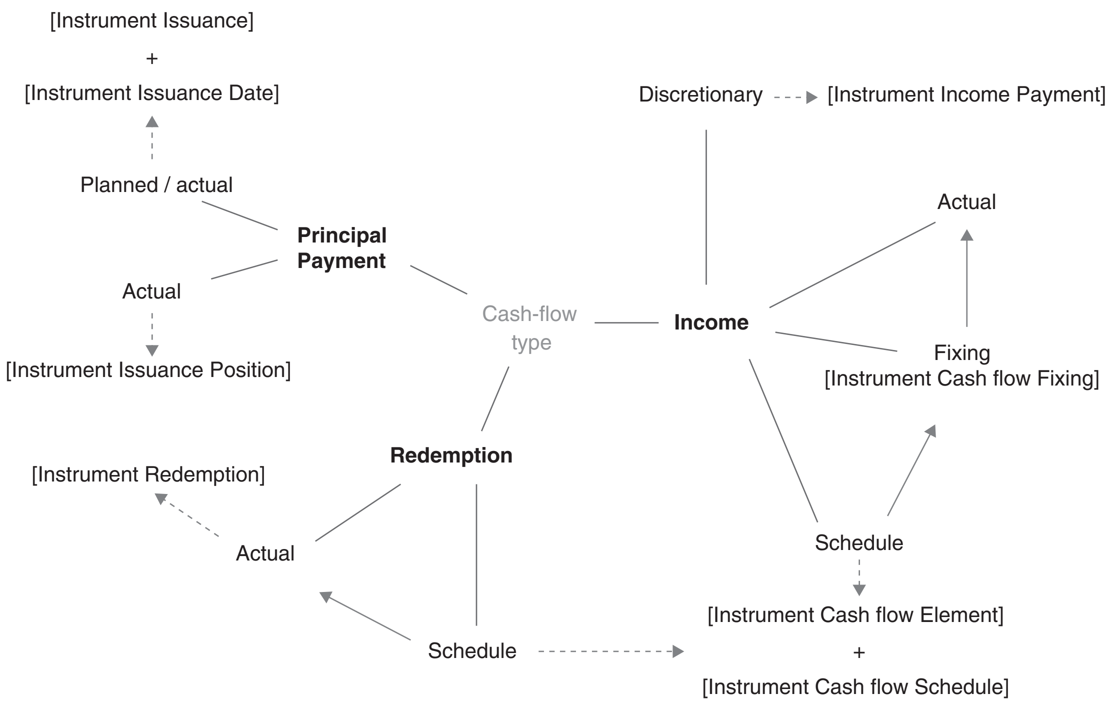


Figure 3.6: Conceptual data model for cash flow information by cash flow type.

Fortunately, the additivity of the monetary cash flows generated by a financial instrument results in a clear dichotomy between cash flow types, so data modelling and analysis can be done on each cash flow type separately.[5]

In the following, mapping rules for various types of cash flow information are given in detail using cash flow type specific subcategories, such as coupon type.

3.1.3.5 Storing of Precalculated Cash Flow Structures

In our model, the place to store such information is the Instrument Cash Flow Fixing class of attributes (see Table 3.11), as prescheduled cash flows can be interpreted as being fixed at issuance.

Table 3.11: Attribute definitions of the instrument *Cash Flow Fixing* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Cash Flow Fixing ID	PK	Unique identifier for an instance of this class. Automatically allocated by the system.	Identifier	M	(Automatic)
Instrument ID	FK	The unique identifier for this instrument data record as allocated by the system. This attribute specifies the instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Cash Flow Element ID		Specifies the element to which an instance of this class belongs.	String	O	—
Cash Flow Fixing Type		Specifies the type of instrument element to which the fixing value applies.	Look-Up	M	Cash Flow Fixing Type
Fixing Date		Specifies the date on which the actual fixing has taken place.	Date	O	—
Effective Date		Specifies the date from which onward the fixed value applies.	Date	M	—
Fixing Value		Specifies the actual value determined for the fixing.	Amount	M	—
Fixing Unit		Specifies the unit in which the fixing is expressed.	Look-Up	M	Unit

[5] This dichotomy is even valid in the case of calculating the theoretical price of an instrument, where the (predominantly used) discounted cash flow method values the various cash flow types separately and then sums over the separate valuations in order to obtain the total value of the instrument.

3.1.3.6 The Structural Approach to Cash Flow Schedule Modelling

When structurally decomposing a cash flow schedule, two aspects need to be distinguished, the timing of the cash flow payments, measured by **cash flow periods**, and the actual amount, measured by the **cash flow amount**.

Cash flow schedules are modelled by cash flow periods. A cash flow period is the time interval in which scheduled cash flow payments become due. In the end of this period, the interest that has been accruing since the settlement of the last coupon is finally paid. It is a defined period in time over which a defined cash flow amount is being paid out. For this cash flow amount, multiple types of schedules may exist, accrual schedule, fixing schedule, and so on. As a consequence, every cash flow period is modelled consisting of

- One instance of Instrument Cash Flow Element determining the cash flow amount of the period.
- One or many instances of Instrument Cash Flow Schedule, depending on how many schedules are defined for this period.

Figure 3.7 illustrates the logic of using the two mentioned tables for the determination of a simple cash flow schedule:

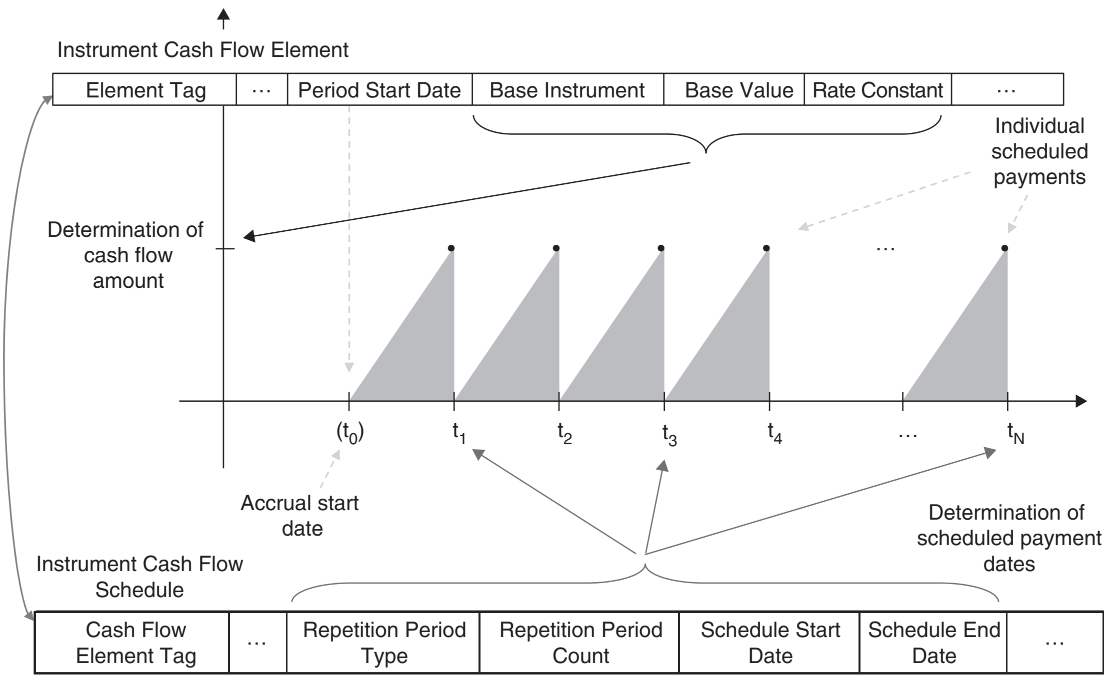


Figure 3.7: Modelling of a simple cash flow schedule with single amount and series of payment dates using the Instrument Cash Flow Element and Instrument Cash Flow Schedule tables.

- **Instrument Cash Flow Element.Period Start Date** specifies the Accrual Start Date of the first coupon period represented by this cash flow element. Accrual start dates for the subsequent income periods correspond to the scheduled payment dates of the previous income period.
- **Instrument Cash Flow Element** specifies the cash flow amount to be paid during the schedule.
- **Instrument Cash Flow Schedule** stores the meta-information that allows you to identify the exact dates on which the cash flow payments are scheduled.
- Relevant instances (entries) in both tables are linked for a given instrument via the attributes **Instrument Cash Flow Element.Element Tag** and **Instrument Cash Flow Schedule.Cash Flow Element Tag**.

The logic applies regardless of the exact cash flow type.

Figure 3.8 shows a slightly more complicated modelling case, where the cash flow amount changes over the lifetime of the schedule:

- As in the previous example, Instrument Cash Flow Schedule still models a continuous series of dates from t_1 to t_N, determining the length of the overall lifetime of the schedule.
- The change in cash flow amount to be paid out is now captured by two instances of the Instrument Cash Flow Element table:

The first instance, starting at Period Start Date = t_0 captures the first cash flow amount via Rate Constant = 1 (plus additional attributes).

The first instance, starting at Period Start Date = t_3 captures the first cash flow amount via Rate Constant = 2.

- The order in which instance of the Instrument Cash Flow Element table is to be applied is determined by the attribute Period Start Date.
- All instances in both tables are still connected via the attributes Instrument Cash Flow Element.Element Tag and Instrument Cash Flow Schedule.Cash Flow Element Tag.

Single, predefined and scheduled cash flow payments, like bullet redemptions, specify the one-off payment by an entry in Instrument Cash Flow Schedule.Schedule Start Date. The attribute Instrument Cash Flow Element.Period Start Date here gives the opportunity to specify the accrual start date for that cash flow.

It is helpful to consider a number of standard examples and how their schematic modelling would look before going into the in-depth details of the cash flow type specific modelling logic in the subsequent chapter.

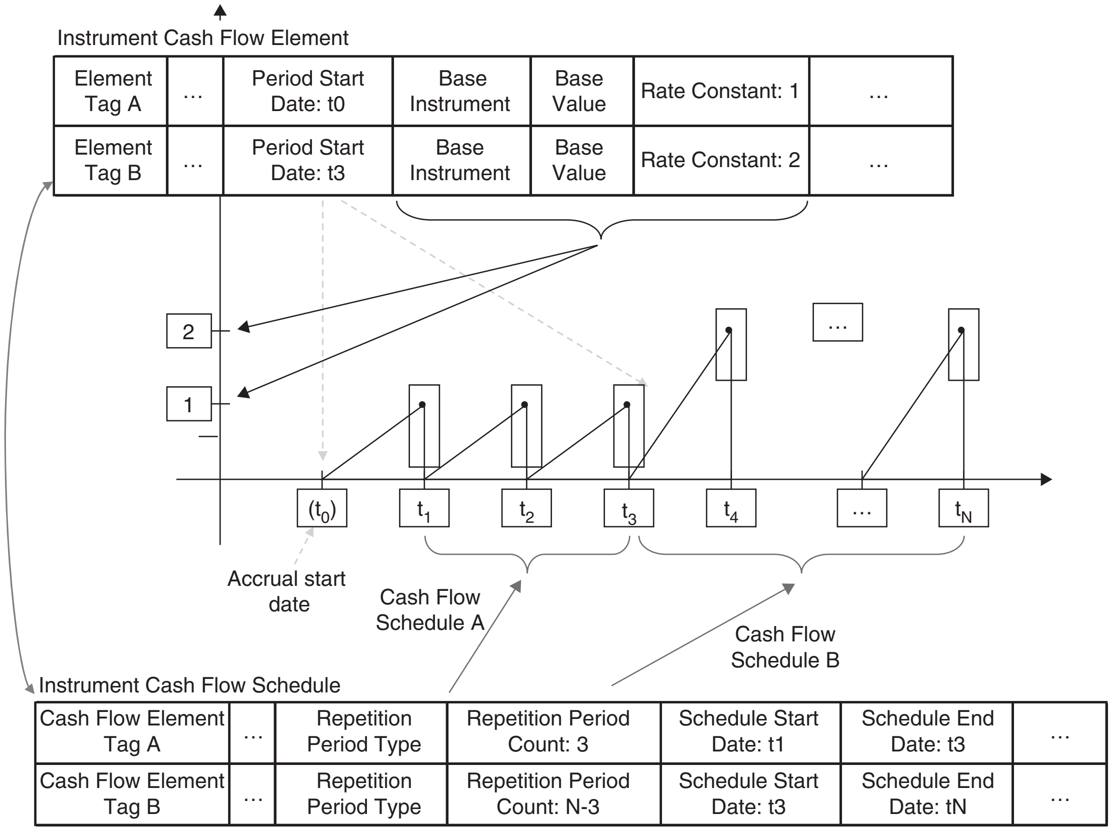


Figure 3.8: Modelling of a stepped cash flow schedule with single amount and series of payment dates using the Instrument Cash Flow Element and Instrument Cash Flow Schedule tables.

- **Accrual schedule** defines the periods over which the income is economically accruing to the investor. The schedule also determines the payout dates.
- **Fixing schedule** defines the date on which the fixing calculation is finished and published. Date from which onward the new fixing applies.
- **Observation schedule** defines the date from which onward makes an observation in the market on which the fixing calculation is being based. Determines the lag between fixing date and observation date.
- **Look-back schedule** defines the period back as an observation schedule for which values are to be aggregated from the fixing calculation.

3.1.3.7 Options Cash Flow Modelling

We can distinguish two types of optionality in financial instruments' cash flows:

A. Cash flows of genuine options; instrument representing a certain right on a specific underlying price or other measure. In that case, the cash flow of these **Option Instruments** is a full cash flow structure in its own right, and the linkage to the underlying is of a mere functional nature. By its very nature, this cash flow structure cannot be reasonably modelled in a precalculated way (although some Data Providers actually try to do so, with interesting results), and we need to resort to our structural modelling setup.

B. **Embedded options**; optional rights to the investor or issuer of an instrument that influences the timing and, in some cases, the size of predefined cash flow schedules. Such rights are intrinsically linked to existing cash flow schedules and, hence, again need to be modelled as add-on cash flow elements to structurally modelled cash flow schedules.

This highlights an important feature of our structural cash flow data model: We can capture information on both types of instruments in the same and unique modelling framework! No separate instrument-type specific attribute classes are needed, not even specific attribute classes for embedded options. It all fits in the general framework outlined in Section 3.1.3.6. This generality pertains to all types of financial instruments. We give examples for the modelling of genuine option instruments in Section 3.1.4.10, and for embedded option modelling in Section 3.1.4.5.

3.1.4 Structural Cash Flow Modelling Examples

We will review the same series of typical examples when we discuss pricing methods in Chapter 4.

3.1.4.1 Zero Coupon Bond: The most simple cash flow

Zero Coupon Bonds or Discounted Debt Instruments are the financial contracts with the simplest cash flow pattern: They involve two separate cash flows (depicted in Figure 3.9):

- An initial principal payment by the investor
- Some future redemption payment by the issuer

The redemption payment to the investor usually consists of the principal amount plus some interest in return for lending the initial amount for the given period. In practice, there is no distinction made between the interest and principal part of the redemption payment, when they are due on the same date.

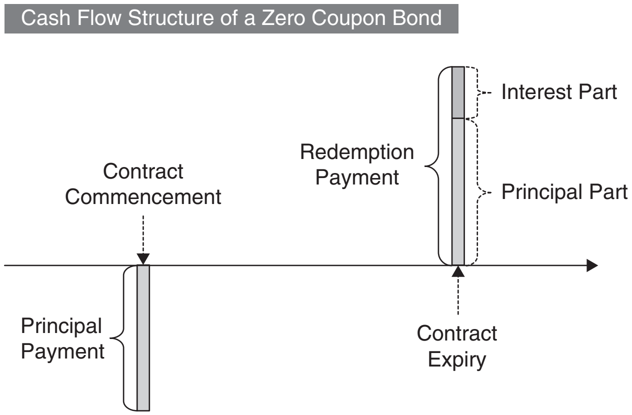


Figure 3.9: Cash flow pattern of a zero coupon or discount debt instrument. Cash flow payments are assumed to coincide with contract commencement (not conclusion!) and expiry.

There is no element of optionality, discretion, or dependence on underlying cash flows or accumulated asset values embedded in this type of contract, and in the absence of credit risk the investor holding the instrument to maturity would have complete certainty as to the cash flow he or she will receive in return for his or her initial investment.

For the individual coupon period 3, Figure 3.10 shows how the interest accrues monotonously over the length of the coupon period: linearly or exponentially. The cash flow is fixed at the beginning and accrues monotonously over the entire period. The cash flow amount to be paid accrues from the beginning of the coupon period and is paid out at the end of it.

Figure 3.11 shows the term sheet of a real zero coupon bond. As in this example all coupon periods have an identical coupon rate specified that is paid at regular intervals; the coupon

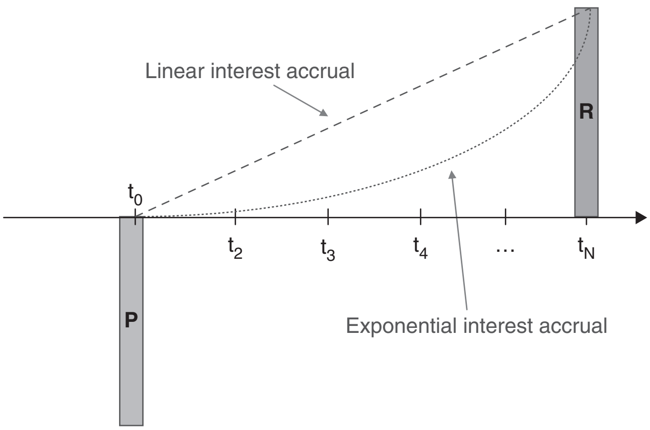


Figure 3.10: Cash flow of a zero coupon bond.

Republic of Argentina
Fixed rate zero coupon
Guaranteed by International Bank for Reconstruction & Development - World Bank
US$ 250.000 m (Cash amount: US$ 235.505 m)
Exchange rate to euro: 1.0699: euro equivalent: 233.667 m (Cash amount: 220.119 m)

Issuer details:-			Guarantor details:-
Type	:	Central government	Supranational institution
Industry sector	:	State Authority/Government	Banking & Financial services
Specific ind. group	:	Government-Central Authorities	Finance-Development Banks/Multilateral Agencies
Primary SIC code	:	General government	Misc. business credit institutions
Primary NAICS code	:	Other General Government Support	Other Nondepository Credit Intermediation
Nationality	:	Argentina	Supranational
Nationality of risk	:	Argentina	

Issue details:-			Issue Ratings:-			
Announced	:	7/10/1999 Registered	Moodys	:	- (Launch)	
Offer date	:	8/10/1999	S & P	:	- (Launch)	- (Current)
Pricing date	:	8/10/1999 Unsubordinated	Fitch	:	- (Launch)	
Payment date	:	15/10/1999	Issue Ratings:-			
Closing date	:	15/10/1999 Denoms (000s): 1	Moodys	:	Ca (Current)	
			S & P	:	B (Current)	
			Fitch	:	DDD (Current)	
Market type	:	Global market public issue				
Pricing:-		(Open)				
Coupon	:	0.00%%%%	Redemption: 100.00%			
Issue price	:	94.202%				
Maturity	:	16/10/2000 (1 year)				
Fees:-			Yields:-		Annual	Semi-Annual
Gross fees	:	0.1250%	Yield to mty	:	6.1373%	6.0459%
			Cost of funds	:	6.2779%	6.1824%

Legal Information:-		
Governing laws	:	New York
Listings	:	Luxembourg
SEC	:	SEC Registered

Other Information:-		
ISIN number	:	US040114BF68
Security number	:	040114BF6

Priced to yield 1-yr Libor minus 0.15%. Simult 5 tranches totalling US$1.25bn.

Figure 3.11: Real world term sheet of a zero coupon bond.

schedule can be summarised in a single coupon stream. Assuming that the interval between two successive coupon payments is exactly 0.5 years, the entire set of coupon periods may be modelled as a single coupon stream (as shown in Table 3.12) and a single redemption stream (as shown in Table 3.13).

3.1.4.2 Straight Coupon Bond: Resuming the real example

The plain vanilla Fixed Coupon Bond or Loan is identical to the Zero Coupon Bond, except that between principal and redemption payment explicit periodic interest payments, so-called coupon payments are made from the debtor (issuer) to the creditor (investor).

Table 3.12: Instrument cash flow schedule table in the zero coupon bond example. Redemption cash flow schedule not strictly needed

Attribute Name	Redemption Schedule
Cash Flow Schedule ID	(Auto)
Cash Flow Element ID	FIX (RED31)
Cash Flow Schedule Type	ACC (Accrual Schedule)
Repetition Period Type	DT (Single Specific Date)
Repetition Period Length	NULL
Schedule Start Date	16.10.2000
Schedule End Date	NULL

Table 3.13: Instrument cash flow element table in the zero coupon bond example

Attribute Name	Redemption Element
Cash Flow Element ID	FIX (RED31)
Instrument ID	(Auto)
Cash Flow Element Type	DRF (Debt Redemption Fixed)
Period Start Date	16.10.2000
Period End Date	NULL
Base Instrument Identifier Type	ISOCCY (Currency)
Base Instrument Identifier	EUR
Base Unit	PRI (Principal)
Base Value	1.0
Base Value Denominator	1.0
Rate Constant	100
Rate Factor	NULL
Cap	NULL
Floor	NULL
Underlying Instrument Identifier Type	NULL
Underlying Instrument Identifier	NULL
Underlying Type	NULL
Underlying Price Type	NULL

Coupon and redemption payments are fixed in amount and timing at the outset of the contract, hence the term Fixed Coupon Bond. Figure 3.12 shows the main conceptual cash flow elements of a straight fixed coupon bond.

Again, there is no element of optionality, discretion, or dependence on underlying cash flows or accumulated asset values embedded in this instrument type, which earns it the term “straight” or “plain vanilla.”

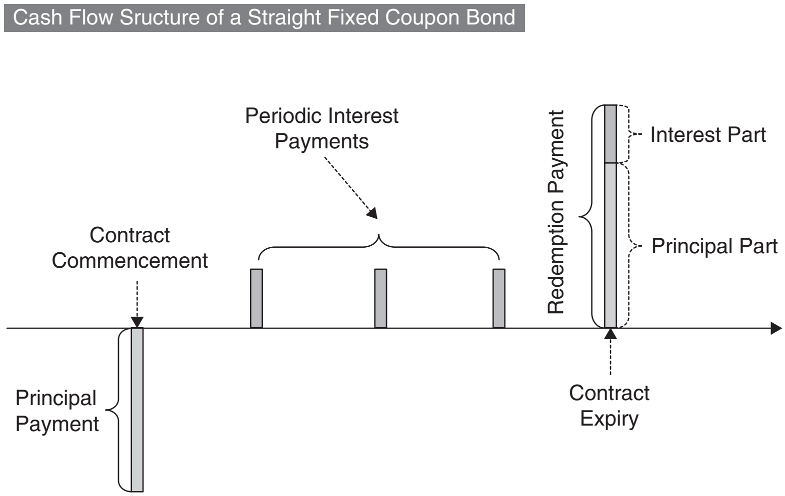


Figure 3.12: Cash flow pattern of a straight fixed coupon bond/straight loan. Cash flow payments are assumed to coincide with contract commencement and expiry.

In order to illustrate these concepts, consider the schematic cash flow of a so-called straight fixed coupon bond debt instrument with bullet (i.e., one-off) redemption payment, depicted in Figure 3.13. The term straight denotes the absence of any embedded optionality potentially affecting the coupon or redemption schedule. According to market conventions, all cash flow elements are expressed in a percent of the nominal amount outstanding.

The bond is issued at date t_0 with an issuance payment of P(%), has subsequent coupon payments of a fixed rate of C(%) at dates t_1, t_2, t_3, and t_4, and has a final (combined) redemption payment of R(%) and coupon payment of C(%) at (maturity) date t_5. In the

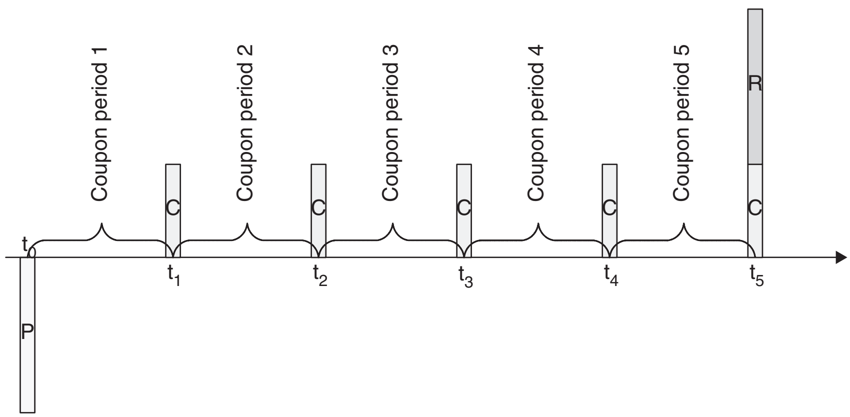


Figure 3.13: Cash flows of a straight fixed coupon bond.

example, the payment date of the last coupon coincides with the redemption date of the principal from the issuer to the investor. With this payment, all claims between issuer and investor are settled.

The income and redemption part of this cash flow structure would be modelled using the (exemplary) entries in the cash flow related tables (Tables 3.14 and 3.15). It is assumed that all payments are scheduled annually and in EUR.

Table 3.14: Instrument cash flow schedule table in the straight fixed coupon bond example

Attribute Name	Coupon Schedule	Redemption Schedule
Cash Flow Schedule ID	(Auto)	(Auto)
Cash Flow Element ID	FIX (CPN12)	FIX (RED31)
Cash Flow Schedule Type	ACC (Accrual Schedule)	ACC (Accrual Schedule)
Repetition Period Type	YR (Period in Years)	DT (Single Specific Date)
Repetition Period Length	1	1
Schedule Start Date	t1	t5
Schedule End Date	t5	t5

Table 3.15: Instrument cash flow element table in the zero coupon bond example

Attribute Name	Coupon Element	Redemption Element
Cash Flow Element ID	FIX (CPN12)	FIX (RED31)
Instrument ID	(Auto)	(Auto)
Cash Flow Element Type	DIF (Debt Income Fixed)	DRF (Debt Redemption Fixed)
Period Start Date	t0	t0
Period End Date	t5	t5
Base Instrument Identifier Type	ISOCCY (Currency)	ISOCCY (Currency)
Base Instrument Identifier	EUR	EUR
Base Unit	PRI (Principal)	PRI (Principal)
Base Value	1.0	1.0
Base Value Denominator	1.0	1.0
Rate Constant	C	100
Rate Factor	NULL	NULL
Cap	NULL	NULL
Floor	NULL	NULL
Underlying Instrument Identifier Type	NULL	NULL
Underlying Instrument Identifier	NULL	NULL
Underlying Type	NULL	NULL
Underlying Price Type	NULL	NULL

3.1.4.3 Reference-linked Debt or Floating Rate Note

Things become a bit more complex when the cash flow amount is not determined at the outset of the financial instrument, but rather varies depending on the evolution of other instruments. Examples are floating and index linked coupons as well as index linked redemptions. In the case of a floating rate instrument, issuers wish to hedge themselves against a fall in interest rates, so they issue the instrument in a way, that for each coupon period, the applicable coupon rate is determined by the level of a reference rate. Hence, instead of one single or a sequence of fixed coupon rates, the prospectus specifies a formula for the calculation of coupon rates applicable for each coupon period, depending on the values of one or more reference rates or indexes in the fixing dates.

In a reference-linked debt instrument either the income cash flow or the redemption cash flow are conditional upon the development of a reference instrument (usually a market-known Interbank rate such as LIBOR or EURIBOR). Examples are Floating Rate Notes (FRN) and Index-Linked Debt.

With income cash flows being dependent on a reference rate, such as in the case of a FRN, the level of the next income cash flow is determined as a function of the reference instrument at regular intervals (Cash Flow Fixings), usually at the time the previous income cash flows are being paid out. Figure 3.14 shows such a pattern with the value of the reference instrument and, hence, the values of the interest cash flows being unknown at the time of contract conclusion (marked as dotted lines).

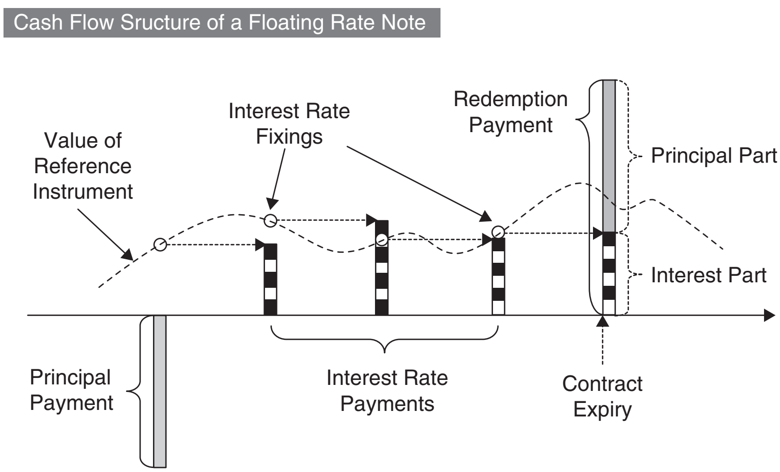


Figure 3.14: Cash flow pattern of a floating rate note. At periodic intervals, income cash flows are determined as a function of a reference instrument. Here it is assumed that this function is a 1 : 1 mapping.

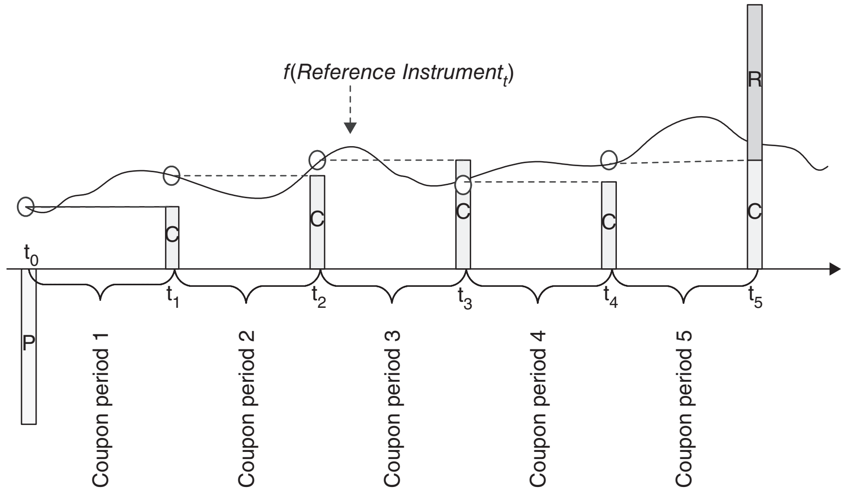


Figure 3.15: Floating rate coupon schedule.

In the example, the simplifying assumption is made that the formula determining the interest cash is a 1 : 1 mapping from the reference instrument. In reality, these functions usually involve at least some mark-up or mark-down (or more complicated formulae) compared to the value of the reference instrument at the cash flow fixing date.

Alternatively, the redemption payment may be depending on the value of the reference instrument at the redemption fixing date.

Another important feature of floating rate coupons is the fixing lag rule. In fact, for some of these instruments, the coupon rate applying to a coupon period is fixed in the beginning of this period, when the interest starts accruing. This is the case depicted in Figure 3.15. In other cases, the coupon rate might be known only at the end of the coupon period.

- Formula is C = a + b(IS123456789123)
- IS123456789123 is the 5-year LIBOR rate in %. The daily closing mid rate is being used.
- Coupon payments are twice per year.

We need two schedules:

- Accrual schedule determining the pay-out dates
- Fixing schedule determining the coupon fixing dates, which can correspond to the pay-out dates of the last fixing, but do not necessarily need to (in our example they do). The modeling of both schedules is illustrated in Tables 3.16 and 3.17.

Table 3.16: Instrument cash flow schedule table in the floating rate note example

Attribute Name	Coupon Schedule 1	Coupon Schedule 2	Redemption Schedule
Cash Flow Schedule ID	(Auto)	(Auto)	(Auto)
Cash Flow Element ID	FLOAT (CPN14)	FLOAT (CPN14)	FIX (RED31)
Cash Flow Schedule Type	ACC (Accrual Schedule)	FIX (Fixing Schedule)	ACC (Accrual Schedule)
Repetition Period Type	MO (Period in Months)	MO (Period in Months)	DT (Single Specific Date)
Repetition Period Length	6	6	NULL
Schedule Start Date	t1	t0	t5
Schedule End Date	t5	t4	t5

Table 3.17: Instrument cash flow element table in the floating rate note example

Attribute Name	Coupon Element	Redemption Element
Cash Flow Element ID	FLOAT (CPN14)	FIX (RED31)
Instrument ID	(Auto)	(Auto)
Cash Flow Element Type	DIV (Debt Income Variable)	DRF (Debt Redemption Fixed)
Period Start Date	t0	t0
Period End Date	t5	t5
Base Instrument Identifier Type	ISOCCY (Currency)	ISOCCY (Currency)
Base Instrument Identifier	EUR	EUR
Base Unit	PRI (Principal)	PRI (Principal)
Base Value	1.0	1.0
Base Value Denominator	1.0	1.0
Rate Constant	+a	100
Rate Factor	+b	NULL
Cap	NULL	NULL
Floor	NULL	NULL
Underlying Instrument Identifier Type	ISIN	NULL
Underlying Instrument Identifier	IS123456789123	NULL
Underlying Type	INS (Single Instrument)	NULL
Underlying Price Type	DCLMID (Daily Close Mid Price)	NULL

3.1.4.4 Step-up Coupon

A stepped or step-up coupon is a debt instrument that changes the level of its coupon payments over time. The classical example is the instrument that pays initially an x% coupon for n years, and then, until maturity, y%. The amount being paid does not need to be fixed up front, but may depend on a variable coupon formula. An example of a combination of a step-up coupon and a floating rate note is given in Figure 3.16.

This is a nice illustration of how all types of scenarios can be and are effectively combined by issuers, in order to achieve their most convenient liability structure. The modeling example of this structure is given in Tables 3.18 and 3.19.

3.1.4.5 Convertible Bond: Semi-fixed event-linked debt

Another type of dependency is making a redemption payment of an instrument fully conditional on whether a certain event occurs. In the case of a Convertible Bond, for example, regular coupon payments are being made, but the redemption payment is usually dependent on the value of a reference instrument being above or below a certain threshold value. If this condition is not met, no redemption payment is being made and the investor is handed out a predetermined number of equity shares (usually the reference instrument) as specified in the contract. Figure 3.17 shows an exemplary cash flow structure for a Convertible Bond.

An analogous setting is the case of credit-linked notes, where the timing of the redemption is made conditional on the event that (some of) the credit contract(s) underlying the note go

Table 3.18: Instrument cash flow schedule table in the step-up coupon example

Attribute Name	Coupon Schedule 1	Coupon Schedule 2	Redemption Schedule
Cash Flow Schedule ID	(Auto)	(Auto)	(Auto)
Cash Flow Element ID	FLOAT (CPN14)	FLOAT (CPN15)	FIX (RED31)
Cash Flow Schedule Type	ACC (Accrual Schedule)	ACC (Accrual Schedule)	ACC (Accrual Schedule)
Repetition Period Type	YR (Period in Years)	YR (Period in Years)	DT (Single Specific Date)
Repetition Period Length	1	1	NULL
Schedule Start Date	25.5.2000	25.5.2005	25.5.2010
Schedule End Date	25.5.2005	25.5.2010	25.5.2010

Westpac Banking Corp
Floating rate note
Step-up coupon
US$ 400.000 m (Cash amount: US$ 399.192 m)
Exchange rate to euro: 0.9113: euro equivalent: 438.933 m (Cash amount: 438.047 m)

Issuer details:-

Type	:	Private bank
Industry sector	:	Banking & Financial services
Specific ind. group	:	Finance-Commercial & Savings Banks
Primary SIC code	:	Commercial banks
Primary NAICS code	:	Commercial Banking
Nationality	:	Australia
Nationality of risk	:	Australia

Issue details:-

Announced	:	2/5/2000	Registered
Offer date	:		
Pricing date	:		Subordinated
Payment date	:	25/5/2000	
Closing date	:	25/5/2000	Denoms (000s): 1

Market type : Euro market public issue

Issue Ratings:-

Moodys	:	A1 (Launch)	
S & P	:	A- (Launch)	- (Current)
Fitch	:	- (Launch)	

Issuer Ratings:-

Moodys	:	Aa3 (Current)
S & P	:	AA- (Current)
Fitch	:	AA- (Current)

Pricing:- (Fixed)

Basis	:	0.45% over 3-min Libor
Accrues from	:	25/5/2000 First payment: 25/8/2000
Issue price	:	99.798% Fixed reoffer price: 99.798%
Maturity	:	25/5/2010 (10 years)
Callable	:	Y

Eff. spread : 48.40 bp

Redemption:100.000%

Call at 100.00% from 25/5/2005 Quarterly

Fees:-

Gross fees	:	0.3500%	Total net fees:	:	0.3500%
Selling conc.	:	0.2000%	Net selling fee	:	0.2000%
Combined mgt•uw	:	0.1500%			

Yields:- Annual Semi-Annual

Legal Information:-

Issued under	:	US$ 12.500.000 m Debt instrument programme
Governing laws	:	England New South Wales
Listings	:	London
Sales restrict	:	United States, United Kingdom, Australia
Rule 144A	:	Eligible
Force majeure	:	Yes
Cross default	:	Issuer - No
Negative pledge	:	Issue - No
Lower Tier II capital		

US sales: Reg S, Cat 2, Tefra D.

Other Information:-

ISIN number	:	XS0112153505, US961214AF08
Security number	:	109802, 109803, 961214AF0
Common code	:	11215360

Step-up coupon = 3-mth Libor • 0.45% until 25/5/2005, thereafter 3-mth-Libor • 0.95%, Lower Tier II capital.

Figure 3.16: Term sheet of a floating rate note step-up coupon.

Table 3.19: Instrument cash flow element table in the step-up coupon example

Attribute Name	Coupon Element 1	Coupon Element 2	Redemption Schedule
Cash Flow Element ID	FLOAT (CPN14)	FLOAT (CPN15)	FIX (RED31)
Instrument ID	(Auto)	(Auto)	(Auto)
Cash Flow Element Type	DIV (Debt Income Variable)	DIV (Debt Income Variable)	DRF (Debt Redemption Fixed)
Period Start Date	25.5.2000	25.5.2005	25.5.2000
Period End Date	25.5.2005	25.5.2010	25.5.2010
Base Instrument Identifier Type	ISOCCY (Currency)	ISOCCY (Currency)	ISOCCY (Currency)
Base Instrument Identifier	EUR	EUR	EUR
Base Unit	PRI (Principal)	PRI (Principal)	PRI (Principal)
Base Value	1.0	1.0	1.0
Base Value Denominator	1.0	1.0	1.0
Rate Constant	0.0000	0.0000	100
Rate Factor	0.4500	0.9500	NULL
Cap	NULL	NULL	NULL
Floor	NULL	NULL	NULL
Underlying Instrument Identifier Type	ISIN	ISIN	NULL
Underlying Instrument Identifier	3-months LIBOR	3-months LIBOR	NULL
Underlying Type	INS (Single Instrument)	INS (Single Instrument)	NULL
Underlying Price Type	DCLMID (Daily Close Mid Price)	DCLMID (Daily Close Mid Price)	NULL

into default. In that case a fully or partial early redemption of the note by the debtor is triggered. Otherwise the note is redeemed at the agreed redemption date.

3.1.4.6 Stock/Equity Shares: Discretionary cash flows

Equity shares represent ownership rights in the assets of a specific company. As such, the purchaser of equity shares is entitled to a proportion of the income and capital of the company. However, the income and capital value cannot be known in advance, and the management of the company has a large amount of discretion in deciding which proportion of income earned will be distributed to the shareholders in the form of dividend at any point of time and which proportion is retained and added to the capital base of the company.

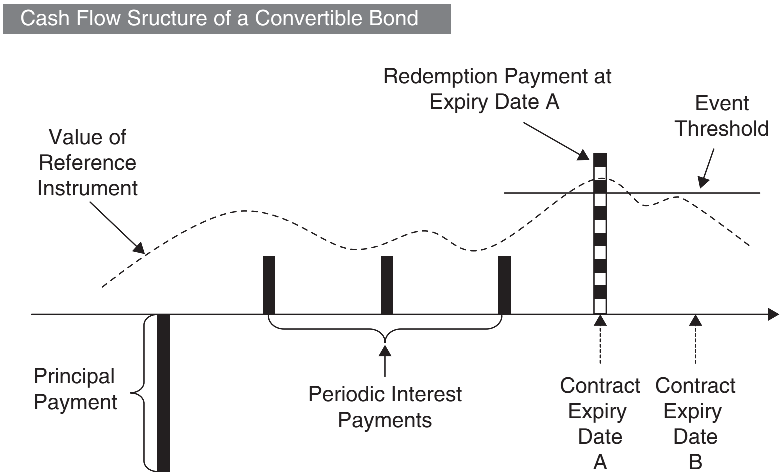


Figure 3.17: Cash Flow Pattern of a Convertible Bond with fixed coupon payments and a conditional redemption payment. The hypothetical value of the redemption payment is shown for two possible redemption dates A and B. Conversion of the value of the reference instrument at the contract expiry date is above the predetermined event threshold value.

A company also has significant freedom as to the terms of different types of shares it issues; that is, offers to investors. Some shares can be given preferential entitlements over income received whereas other types, in certain circumstances, may be designed never to receive any income in return for exclusive rights to the residual capital of the company at a certain date.

The cash flows resulting from equities are therefore dependent on the capacity of a firm's assets to generate income and the value of the assets if and when the company is wound up and its capital distributed to those shareholders entitled to it. Figure 3.18 shows the stylized cash flow pattern for an equity share.

No structural cash flow modelling is needed. All actual income payments are stored as dividends in the *Instrument Payment* class of attributes.

3.1.4.7 Fund Units: Semi-discretionary

Fund units are either equity shares in a fund or entitlements in a fund established as a trust rather than a company. In either case the main difference from equity shares is that the underlying asset pool is not a business but rather a portfolio of financial instruments. The income from the asset pool and its value if the fund is wound up is dependent on the returns from and market prices of the instruments held by the fund.

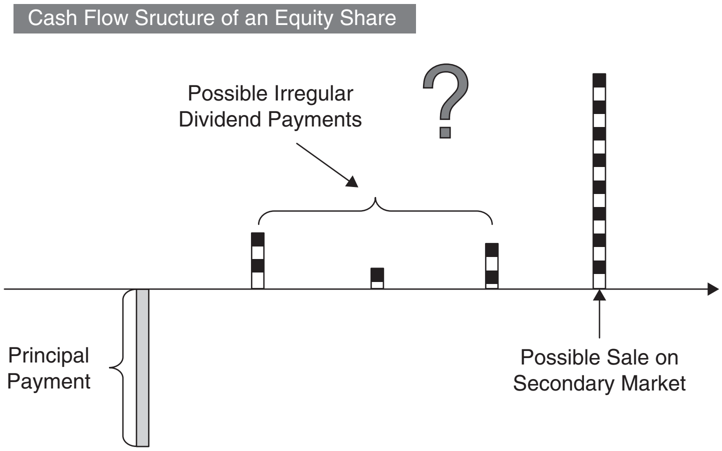


Figure 3.18: Cash flow pattern of an investment in an equity share.

In addition, the investors in the fund usually have little discretion as to the proportion of realised income to pay back to investors, and may not even have any discretion as to when the fund is wound up and its assets are realised and the resulting cash flow distributed.

Hence, the cash flow pattern from an investment in a fund unit is almost identical to that of an investment in an equity share.

3.1.4.8 Forward and Future Contracts

A forward contract is a purchase, sale, or loan taking place some time in the future but with terms agreed today. Figure 3.19 shows a typical cash flow pattern of a forward contract.

A Forward Foreign Exchange (FFX) for instance is an agreement to purchase or sell a certain amount of a given currency in return for payment in another currency at a preagreed exchange rate where the payments between buyer and seller, respectively, are to be settled at a future date.

A Forward Rate Agreement (FRA) is an agreement to lend or borrow a given amount of money for a given period at a given interest rate where the agreed loan period starts at some date in the future. In contrast to the FFX however, the FRA normally does not result in one party actually providing the agreed loan. Instead at the agreed settlement date the parties settle the difference between the market rate for loans of the same duration and the preagreed interest rate for the FRA. This practice is called cash settlement and is a common feature of many types of Forward, Future Swap, and Option contracts.

Note that the Forward Price is not to be confused with the price for entering into the forward contract, commonly called the Forward Rate.

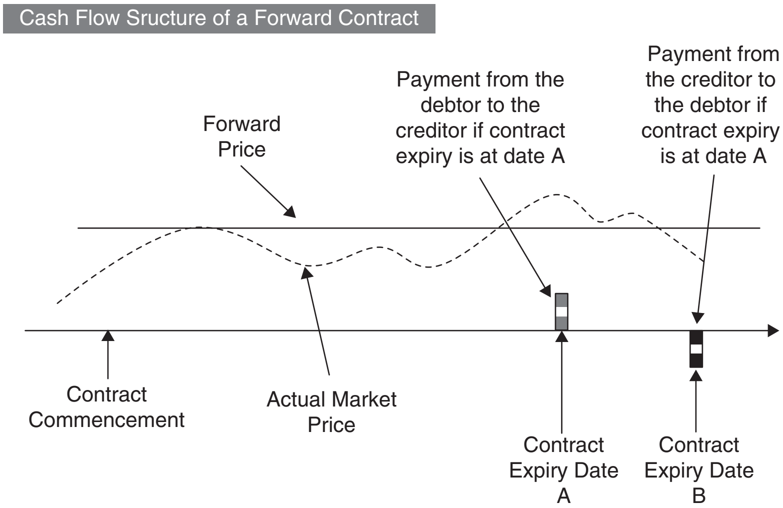


Figure 3.19: Cash flow structure of a forward contract.

Futures are similar to forward contracts but differ in two respects:

1. Forwards are individually negotiated contracts between two parties; Futures are traded on an exchange with standardised contracts.
2. Instead of settling the transaction at the end when it becomes due, a Future is marked-to-market every day, and any resulting loss or profit is settled via the exchange's clearinghouse on a daily basis.

Figure 3.20 shows the resulting stylized cash flow pattern. No structural cash flow modelling is needed. The only aspect to store is the agreed future price.

3.1.4.9 Swap Contracts: Changing the odds

Swaps are also similar to forwards in that they involve transaction some time in the future at terms agreed today. Like forward contracts, swaps normally are negotiated individually rather than traded on an exchange. However, whereas forwards involve a purchase, sale, or loan, a swap involves the exchange of two payment streams between the two parties involved.

An Interest Rate Swap (IRS) normally involves one party paying a pre-agreed Fixed Interest Rate on a notional amount of money to the other party while receiving interest on the same notional amount of money based on floating interest rate such as LIBOR (London Inter Bank Offer Rate), EURIBOR, or other market interest rates. Figure 3.21 gives an example of a typical Interest Rate Swap.

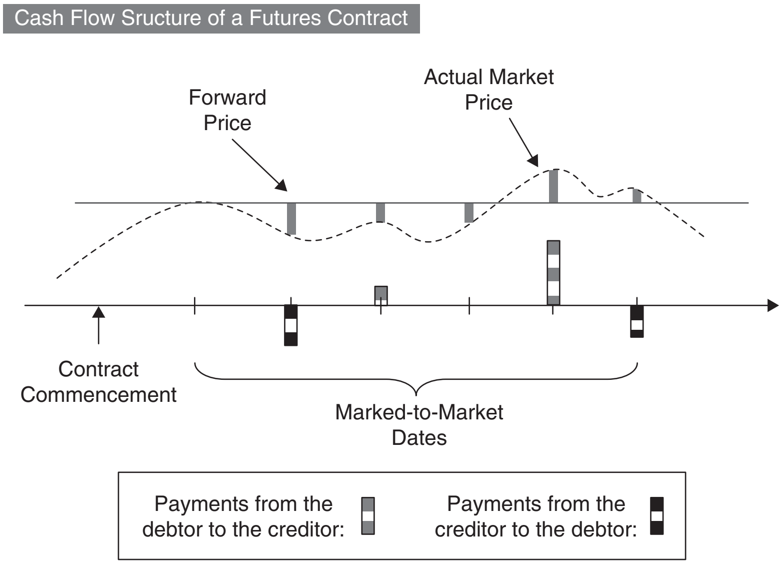


Figure 3.20: Cash flow pattern of a futures contract that is marked-to-market at regular intervals.

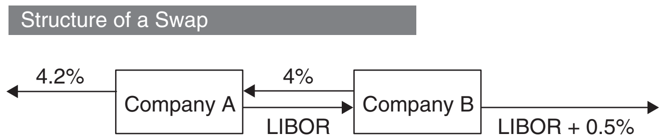


Figure 3.21: Companies A and B use a swap to transform a liability of A *vis-à-vis* a third party.

3.1.4.10 Options and Warrants: Taking the odds

Options and warrants in their basic form are rights but not obligations that entitle the holder of the right to sell or purchase a certain quantity of a given asset at a point in time or a given period for some preagreed price. To the writer of the option they are obligations to complete the preagreed purchase or sale at the discretion of the holder of the rights under the option contract.

The asset involved can be a real asset such as a commodity, a financial asset such an equity, or a notional asset that is constructed from other reference quantities such as indices, inflation, or interest rates.

Like forwards, futures and swaps options, and warrants, the actual cash flow pattern for options is dependent on one or more reference quantities such as real asset prices, financial asset prices, or other reference quantities such as interest rates, exchange rates, commodity prices, or inflation rates.

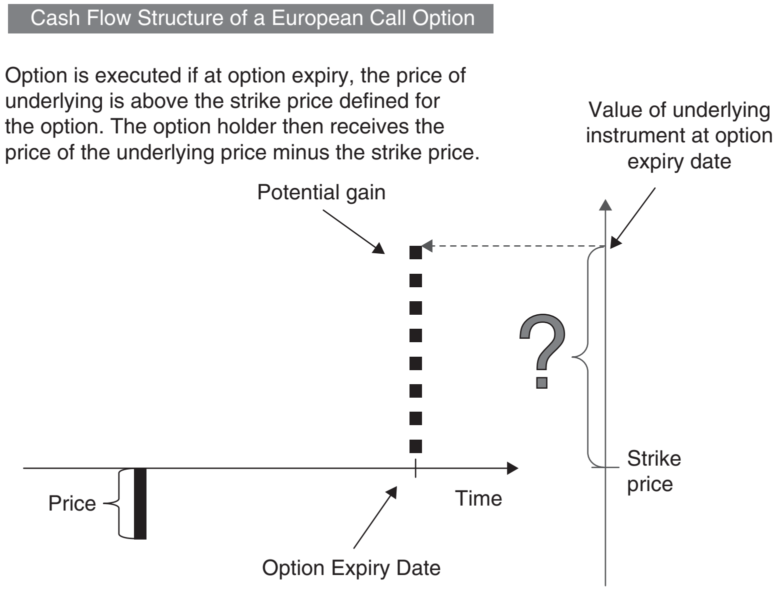


Figure 3.22: Cash flow structure of a European call option.

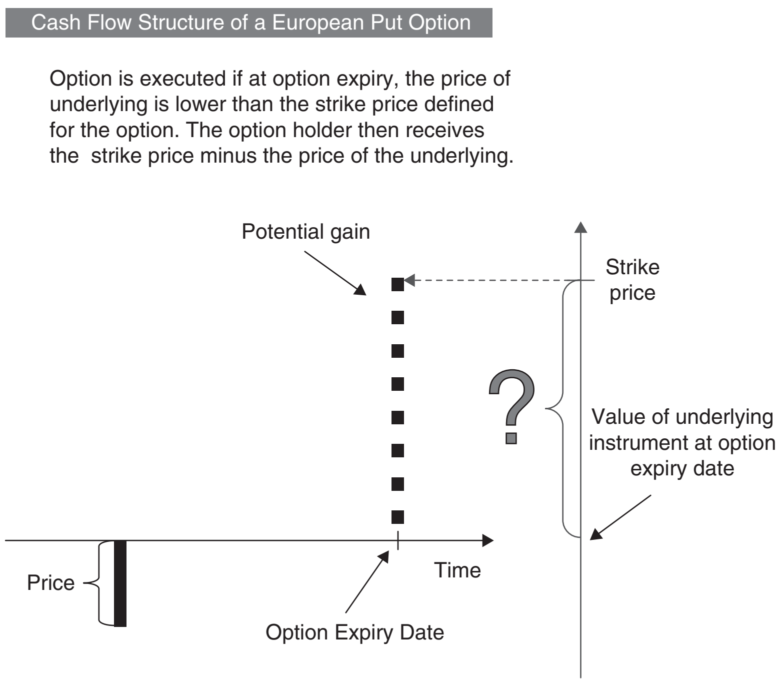


Figure 3.23: Cash flow structure of a European put option.

Options are distinguished largely by two types of features:

- Whether they give the right to sell (put option) or buy (call option) the underlying at a prefixed price (strike price)
- When the execution of the right may take place:
  - European options allow the execution only at the option expiry date.
  - Bermudan options allow the execution at a series of prefixed dates.
  - American options allow the execution at any point during the lifetime of the option.

For convenience, we restrict our modelling discussion to European Options only. Figures 3.22 and 3.23 show the corresponding cash flow patterns. The extension to Bermudan and American options is straightforward—at least from a modelling perspective: Just replace the option expiry date by the next possible execution date.

CHAPTER 4

Introduction to Practical Valuation

In this chapter we will take a first look on how different instruments can be priced using a relatively small set of tools. We present the most commonly used approaches and solutions, but do not enter into the pros and cons of each method. If you wish to follow market standards you can safely apply the presented toolbox. However, if you wish to overcome known shortcomings and "beat the market," you are advised to study the related academic literature. Use Cochrane (2005) as a starting to that literature.

We start with pricing using discounted cash flows, which is arguably the most fundamental tool. We then move on to market interest rates and in particular how to use yield curves to obtain appropriate estimates for market interest rates for different horizons.

Next we take a brief look at "beta" calculation and the cost of equity before moving on to option pricing. Finally we introduce the most popular building blocks for financial instruments that should be in any valuation toolbox: a basic set of commonly used pricing models covering debt instruments, equity, and simple options.

Each tool description is accompanied by a hands-on exercise in our companion web site toolbox.

4.1 The Basic Tools for Valuation

The starting point for our journey will be some basic knowledge of the most fundamental and common tools for valuation. In the first part of the chapter we take a look at the pricing with discounted cash flows, which is arguably the most fundamental tool. Next we look at market interest rates and in particular how to use yield curves to obtain appropriate estimates for market interest rates for different horizons.

Next we will take a brief look at the calculation of beta and the cost of equity. The last tool we will look at in the first section is option pricing. This is second only to discounted cash flow valuation in its universal applicability and is completely indispensable for a very large number of valuation tasks.

4.1.1 Pricing with Discounted Cash Flows

Anybody who has cash available today and is willing and able to lend or invest normally would do so only if there was some reward. This reward might be a pre-agreed fixed percentage of the amount on loan or invested or some agreed share of rewards from a business or other asset. In the simpler cases we may know the actual amounts of the future rewards in monetary terms already, whereas in the more complex cases we may know the actual amounts only later.

In the simpler cases we can usually directly work out the price using the Discounted Cash Flow Valuation methods and an appropriate interest rate. In more complex cases, as long as no optionality is involved we can still use Discounted Cash Flow Valuation methods, albeit in some modified form.

The diagram in Figure 4.1 shows a simple loan or investment where both the time of repayment of the amount invested plus the pre-agreed reward and its actual amount are known. The cash payment at t_0 is the price of the future cash flow at t_1 in today's (t_0) money, and the cash flow at t_1 is the original amount on loan plus some amount of interest. If the loan is for exactly one year then the ratio of the reward to the amount on loan is the annual interest rate.

If instead we know the interest rate to use and also the amount we will receive at the end of the loan we can similarly calculate the value of the cash flow in today's money. This value in today's money is usually called **Present Value** and in fact is the price of the future cash flow today. Formula 4-1 provides a general method for calculating the price or present

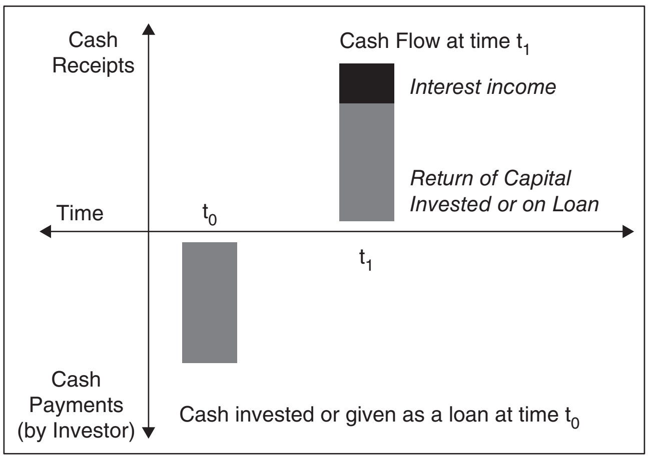


Figure 4.1: Cash flow timeline for a simple investment or loan.

value of a cash flow that an investor will receive at some arbitrary but pre-agreed time in the future given an appropriate market interest rate.

Formula 4-1: Valuation using annual compounding and an annual interest rate

$$Price_{CashFlow} = PV_{CashFlow} \frac{F}{(1 + r_{annual})^{-T}}$$

where:

F: face value of the instrument

T: time to maturity in number of years

r_{annual}**:** annual interest rate applicable

Formula 4-1 assumes that interest is calculated and added to the loan amount once a year and the resulting compound value is then used as the basis for the following year's calculation for interest due. This is called **annual compounding**. If we have a nominal annual interest rate but actually compound the interest more frequently, for instance every month, we first have to translate the nominal annual interest rate into one that is adjusted for the effect of more frequent compounding. Formula 4-2 provides the method by which we can translate a nominal annual interest rate into one that is adjusted for the effects of compounding over a period other than a year.

Formula 4-2: Translating an annual rate into a rate for other compounding periods

$$r_{effective} = [1 + (r_{annual}/n)] - 1$$

where:

$r_{effective}$**:** the equivalent effective annual interest rate for a given compounding period n

r_{annual}**:** the nominal annual interest rate

n: the number of the periods equivalent to exactly one year

For example, if the period is 6 months, representing a semi-annual frequency, n = 2. If the period is one week, n = 52, and so on.

Often it is better to use calculations that assume **continuous compounding**. Continuous compounding is the case in which the time period for compounding is infinitely small and thus continuous. If we have a nominal annual interest rate and any given period for compounding we can use Formula 4-3 to translate it into an equivalent interest rate for calculations assuming continuous compounding.

Formula 4-3: Translating an annual rate into one using continuous compounding

$$r_{continuous} = \ln\left(\left[1 + \left(r_{annual}/n\right)\right]\right) * n$$

where:

$r_{continuous}$**:** the annual interest rate for continuous compounding

r_{annual}**:** the nominal annual interest rate

n: the number of the periods equivalent to exactly one year

ln( ): natural logarithm that is the logarithm with base e

If the interest rate we have is a continuous compounding interest rate, then we need to use a modified form of the discounted cash flow formula given in Formula 4-1. This modified discounted cash flow valuation formula for continuous compounding interest rates is shown in Formula 4-4.

Formula 4-4: Valuing a cash flow using a continuous compounding interest rate

$$Price_{CashFlow} = PV_{CashFlow} = \frac{F}{e^{Tr}} = Fe^{-Tr}$$

where:

$PV_{CashFlow}$**:** the present value or price of future cash flows in today's money

F: cash flow amount to be received at time t in the future

T: the time to maturity, i.e. cash flow payment date, in number of years

r: the continuously compounded annual interest rate for maturity T

e: the exponential constant

We will later see that interest rates are normally not the same for different periods and thus we normally need to calculate the price of each cash flow and then add them all together if we want to get a price for an asset that has more than one cash flow (for instance, a bond paying coupon as well as returning the principal amount invested at maturity).

Sometimes, however, we might want to have a quick approximation for the price of such an instrument using just one average interest rate for all cash flows of the instrument. Formula 4-5 provides such an approximation for calculating the price for a fixed coupon bond as illustrated by the cash flow timeline diagram in Figure 4.2.

Formula 4-5: Valuing a Coupon Bond using a continuous compounding interest rate

$$\text{Price}_{\text{CouponBond}} = \frac{F_{\text{Coupon}}}{e^{r}-1} \times \left[1 - \frac{1}{e^{Nr}}\right] + \frac{F_{\text{Principal}}}{e^{Nr}}$$

where:

$PV_{CashFlow}$: the present value or price of a future cash flow in today's money

F_{Coupon}: coupon cash flows from the bond

$F_{Principal}$: cash flows from repayment of the principal of the bond

T: the time to maturity in number of years

n: number of coupon payments per year

N = T × [1/n)

r: the continuously compounded average annual interest

e: the exponential constant

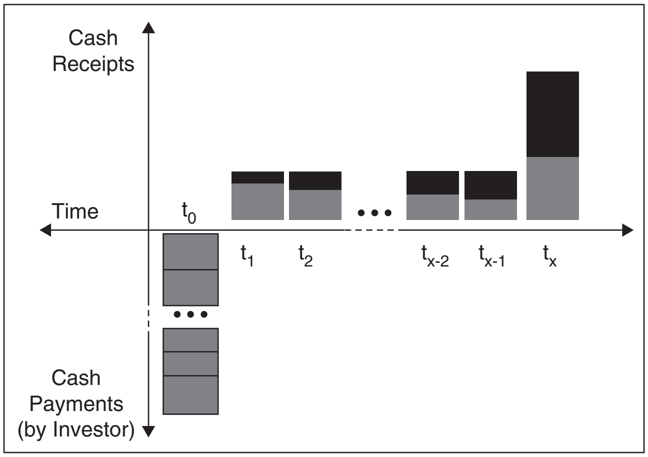


Figure 4.2: Cash flows of a coupon bond.

There are two common variations of the theme of valuing a coupon bond: First the valuation of an annuity, which is the same as valuing a coupon bond without including the value of the repayment of the principal at redemption. The cash flow timeline in Figure 4.3 illustrates this well since it is essentially the same as the cash flow time for a coupon bond as shown in Figure 4.2. A perpetuity, as illustrated by Figure 4.4, is an annuity with an infinite end date and thus again simply a special case of the valuation formula for a coupon bond.

Formula 4-6 shows how to value an annuity. You will note that this is the same as Formula 4-5 except that the final term for calculating the value of the principal repayment has been omitted. Formula 4-7 shows how to calculate the price of a perpetuity.

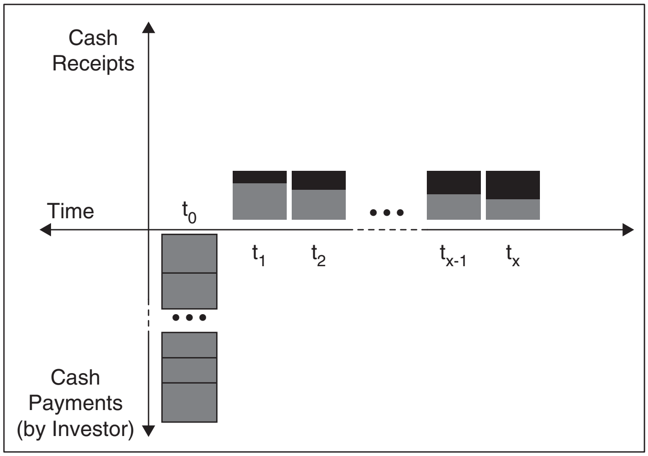


Figure 4.3: Cash flows of an annuity.

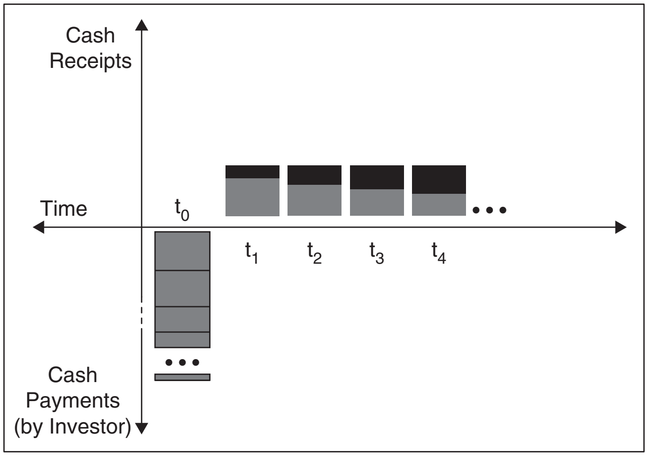


Figure 4.4: Cash flows of a perpetuity.

In this case the formula is similar to both Formulas 4-5 and 4-6, with the final and the penultimate terms missing. The final term is missing because there is no repayment of principal, and the penultimate because the value of the second term becomes 1 as the number of payments increases.

Formula 4-6: Valuing an annuity using a continuous compounding interest rate

$$Price_{Annuity} = \frac{F_{Annuity}}{e^{r} - 1} \times \left[1 - \frac{1}{e^{Nr}}\right]$$

where:

$PV_{CashFlow}$: present value or price of future cash flows in today's money

$F_{Annuity}$: coupon cash flows from the bond

T: time to maturity in number of years

n: number of annuity payments per year

N = T × [1/n)

r: continuously compounded annual interest

e: exponential constant

Formula 4-7: Valuing a perpetuity using a continuous compounding interest rate

$$Price_{CouponBond} = \frac{F_{Annuity}}{e^{r} - 1}$$

where:

$F_{Annuity}$: coupon cash flows from the bond

T: time to maturity in number of years

n: number of annuity payments per year

r: continuously compounded annual interest

e: exponential constant

In order to improve your intuition about these formulas and their logic, it is helpful to take the time working through Lab Exercise 4.1 below.

Lab Exercise 4.1: Discounted Cash Flow Valuation

1. Open the file **DCfValuation.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the valuation calculation shown above.
2. Using the R application load the file **DCfValuation.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseDCfValuation()

...

R> RunDCfValuation()

...

R> DCfValuation()
```

4.1.2 Interest Rates and Yield Curves

So far, we have taken the interest rates used in discounting for granted assuming that we can simply obtain an appropriate market interest rate for a given term from somewhere. In practice, however, obtaining such an appropriate interest rate is often one of the key challenges that must be overcome in the valuation process.

4.1.2.1 Zero Bond Rates and Interpolation

The simplest way to obtain an appropriate interest rate for discounting a given future cash flow would be to find a comparable zero coupon instrument with a redemption date that falls on the same date as the cash flow. In general it is unlikely that such an instrument would be available for an arbitrary date. Fortunately this can be easily overcome. If no zero coupon instruments with a fitting date exists we can use interpolation to arrive at an approximate rate from one zero coupon instrument that matures before the cash flow we want to price and one that matures after it. Formula 4-8 provides a method for Linear Interpolation that you can use to calculate an approximate rate for a given cash flow date that lies between the maturity dates of two zero coupon instruments such as either Zero Coupon Bonds or money market instruments that do not pay coupons (such as commercial paper).

Formula 4-8: Linear Interpolation

$$r_i = r_1 + \left[(r_2 - r_1) \times \frac{(t_i - t_1)}{(t_2 - t_1)} \right]$$

where:

r_i: interest rate for the cash flow for which we need an interest rate

t_i: time to the cash flow for which we need an interest rate in number of years

t_1: time to maturity in number of years for the zero coupon instrument maturing before the cash flow

t_2: time to maturity in number of years for the zero coupon instrument maturing after the cash flow

r_1: interest rate for the zero coupon instrument maturing before the cash flow; this is the yield to maturity for the instrument that is simply the Internal Rate of Return (IRR) of the instrument given its current market price and pre-agreed redemption value

r_2: interest rate for the zero coupon instrument maturing after the cash flow

In Example 4.1 you can see how Formula 4-8 can be used with some specific zero coupon bonds to obtain an interpolated interest rate for a point in time that lies between the maturity dates of the two chosen reference bonds.

Example 4.1: Linear Interpolation

Assume you have a cash flow in 4.25 years for which you need an interest rate. You have also already identified two zero coupon bonds, one with a maturity of three years and one with a maturity of six years from which you can interpolate the rate for 4.25 years. The bond with three years to maturity has a yield to maturity of 3.88% and the 6 year bond has a yield to maturity of 4.13%. Using Formula 4-8 you can then calculate

$$r_i = r_1 + \left[(r_2 - r_1) \times \frac{(t_i - t_1)}{(t_2 - t_1)} \right]$$

$$= 3.88 + \left[(3.88 - 4.13) \times \frac{(4\text{-}25 - 3.00)}{(6.00 - 3.00)} \right]$$

$$= 3.88 + \left[(3.88 - 4.13) \times \frac{(4\text{-}25 - 3.00)}{(6.00 - 3.00)} \right]$$

$$= 3.88 + (-0.25) \times \frac{1.25}{3.00}$$

$$= 3.88 + 0.11$$

$$= 3.99\%$$

Thus, the interpolated interest rate for a cash flow in 4.25 years would be $\mathbf{r_i = 3.99\%}$.

4.1.2.2 Bootstrapping and Interpolation

Although in some cases you may have two or more suitable Zero Coupon Bonds for which you may be able to obtain accurate market prices, in general there are not enough zero coupon instruments to be able to construct complete yield curves from the shortest maturities all the way to long-term maturities above 10 years and reaching 15, 20, or even 30 years.

One way to get around this problem is to recover zero coupon rates from the prices of coupon bearing bonds using a procedure called **bootstrapping**.

To illustrate what bootstrapping is, let's consider the case where you have obtained a collection of coupon bonds each paying a six monthly coupon. The shortest maturity in this collection is a bond with only the last coupon and the redemption still outstanding, and thus six months or less from maturity. The next maturity in the collection should be a bond with two coupons still outstanding and thus a year or less from redemption. The remaining bonds in the collection in increasing order of time to maturity each should have one more additional coupon period remaining the one preceding it, starting with one that has three coupons outstanding.

Bootstrapping works by solving first for the interest rate implied by the price for the bond with only one coupon remaining. The interest rate obtained in this way can then be substituted back into the calculation for the implied interest rate for the last coupon and redemption of the bond with two coupons outstanding, which now becomes solvable.

After solving for the implied one year zero coupon interest rate we can then substitute this together with the six month rate back into the calculation for the implied interest rate for the last coupon and redemption of the bond with three coupons outstanding. This will let us then solve for the zero coupon interest rate applicable to a term of 1.5 years.

If we now continue in the same way we will be able to obtain a zero coupon rate for each maturity in the set. Formula 4-9 provides a method for performing the bootstrap calculation in one step using matrix notation. Denote a discount factor d(t) as a function of a term t specific interest rates r(t) as

$$d(t) = 1/(1 + r(t))^t$$

Formula 4-9 then shows that a vector of discount factors **d** multiplied by the matrix of coupon and redemption cash flows **F** results in the vector **p** of observed bond prices. This can be rearranged to solve for discount factors in **d**, which can be obtained by multiplying the transpose of Matrix **F** with the vector **p**. This method is implemented in the code for Lab Exercise 4.2 on page 89.

Formula 4-9: Calculating zero bond interest rates by bootstrapping

In matrix notation the problem is formulated as follows:

$$\mathbf{d} \times \mathbf{F} = \mathbf{p}$$

In full matrix notation the formulation looks as follows:

$$\begin{pmatrix} d(t_1) \\ d(t_2) \\ \vdots \\ d(t_n) \end{pmatrix} \begin{pmatrix} F(t_1,1) & & & \\ F(t_2,1) & F(t_2,2) & & \\ \vdots & \vdots & \ddots & \\ F(t_n,1) & F(t_1,2) & \cdots & F(t_n,n) \end{pmatrix} = \begin{pmatrix} p(t_1) \\ p(t_2) \\ \vdots \\ p(t_n) \end{pmatrix}$$

Solving for **d** we can rearrange as follows:

$$\mathbf{p} \times \mathbf{F}^{\mathbf{T}} = \mathbf{d}$$

or in matrix notation:

$$\begin{pmatrix} p(t_1) \\ p(t_2) \\ \vdots \\ p(t_n) \end{pmatrix} = \begin{pmatrix} F(t_1,1) & & & \\ F(t_2,1) & F(t_2,2) & & \\ \vdots & \vdots & \ddots & \\ F(t_n,1) & F(t_1,2) & \cdots & F(t_n,n) \end{pmatrix}^T = \begin{pmatrix} d(t_1) \\ d(t_2) \\ \vdots \\ d(t_n) \end{pmatrix}$$

where:

p: vector of observed market prices $\mathbf{P(t_1) \ldots P(t_n)}$

F: matrix of coupon and principal cash flows $\mathbf{F(t_1,1) \ldots F(t_n,1)}$

$\mathbf{F^T}$**:** transpose of the matrix of coupon and principal cash flows

d: vector of bootstrap zero coupon interest rates $\mathbf{r(t_1) \ldots r(t_n)}$

$\mathbf{p(t_1) \ldots p(t_n)}$**:** observed market prices of coupon bearing bonds with a range of maturities from, say, t_1 = 6 months to $t_n = n \times 6$ months

$\mathbf{F(t_1,1) \ldots F(t_n,1)}$**:** coupon and principal cash flows of coupon bearing bonds with a range of maturities from, say, t_1 = 6 months to $t_n = n \times 6$ months at times t_1 = 6 months to $t_n = n \times 6$ months

$\mathbf{r(t_1) \ldots r(t_n)}$**:** calculated bootstrap zero coupon interest rates for a range of maturities from, say, t_1 = 6 months to $t_n = n \times 6$ months

Nota Bene. The above method assumes that the series of cash flow dates coincide between the instruments used. This is a general shortcoming of the Bootstrapping Method. For the method to work, real cash flows have to be properly adjusted to fit the generic cash flow schedule used in the method.

Once we have obtained a set of zero coupon rates we can again interpolate rates between the maturities of the set of zero coupon instruments we used as input. We could again use linear interpolation. Alternatively we can use cubic interpolation, which will result in a smoother overall curve. Formula 4-10 provides a method for cubic interpolation that we can use for calculating intermediate points either in order to obtain rates for use in discounted cash flow valuation or to draw the full yield curve.

Formula 4-10: Cubic interpolation of interest rates

$$r_i = r_1 \times \lambda_1 + r_2 \times \lambda_2 + r_3 \times \lambda_3 + r_4 \times \lambda_4$$

using

$$\lambda_1 = \frac{(t_i - t_2) \times (t_i - t_3) \times (t_i - t_4)}{(t_1 - t_2) \times (t_1 - t_3) \times (t_1 - t_4)}$$

$$\lambda_2 = \frac{(t_i - t_1) \times (t_i - t_3) \times (t_i - t_4)}{(t_2 - t_1) \times (t_2 - t_3) \times (t_2 - t_4)}$$

$$\lambda_3 = \frac{(t_i - t_1) \times (t_i - t_2) \times (t_i - t_4)}{(t_3 - t_1) \times (t_3 - t_2) \times (t_3 - t_4)}$$

$$\lambda_4 = \frac{(t_i - t_1) \times (t_i - t_2) \times (t_i - t_3)}{(t_4 - t_1) \times (t_4 - t_2) \times (t_4 - t_3)}$$

where:

r_i: interpolated interest rate we are looking to obtain to use for pricing given the cash flow

t_i: time to the cash flow for which we need an interest rate in number of years

t_1: time to maturity in number of years for the zero coupon instrument maturing before the cash flow and before **t_2**

t_2: time to maturity in number of years for the zero coupon instrument maturing before the cash flow but after **t_1**

t_3: time to maturity in number of years for the zero coupon instrument maturing after the cash flow but before **t_4**

t_4: time to maturity in number of years for the zero coupon instrument maturing after the cash flow and after **t_3**

r_1: interest rate for the zero coupon instrument maturing at **t_1**

r_2: interest rate for the zero coupon instrument maturing at **t_2**

r_3: interest rate for the zero coupon instrument maturing at **t_3**

r_4: interest rate for the zero coupon instrument maturing at **t_4**

λ_1: interpolation factor for R_1
λ_2: interpolation factor for R_2
λ_3: interpolation factor for R_3
λ_4: interpolation factor for R_4

Again, it is helpful to let the above logic sink in using the Lab Exercise below of the companion website.

Lab Exercise 4.2: Bootstrapping and Interpolating Zero Coupon Interest Rates

1. Open the file **YCurveBootstrap.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the yield curve calculation shown above.
2. Using the R application load the file **YCurveBootstrap.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseYCurveBootstrap()

...

R> RunYCurveBootstrap()

...

R> YCurveBootstrapReports()
```

4.1.2.3 Parametric Yield Curve Estimation

Although the bootstrap method is easy to implement, it has one important drawback: Since it fits the resulting yield curve exactly to the data we have used as input it does not deal well with the errors that are unavoidable in the prices used. Bootstrapping will thus tend to be significantly influenced by any errors in individual bond prices. To alleviate this problem we can use a parametric yield curve estimation approach that fits a suitable parametric function to the observed data using a Nonlinear Least Squares estimation algorithm.

One of the earliest such approaches is the method set out by Nelson & Siegel (1987). Formula 4-11 is implemented in Lab Exercise 4.3, Parametric Yield Curve Estimation, which uses a Nonlinear Least Squares estimation algorithm to fit this formula to a given data set of fixed income instruments.

Formula 4-11: Nelson & Siegel (1987) parametric yield curve estimation

$$p(t_x) = \sum_{k=1}^{n} F_k \exp\left\{-\alpha_1 t_k - \beta(\alpha_2 + \alpha_3)\left(1 - e^{-t_k/\beta}\right) + \alpha_3 t_k e^{-t_k/\beta}\right\}$$

where:

$p(t_x)$: observed market price of a bond with time to maturity $\mathbf{t_x}$

$F_k(t_x)$: k[th] (coupon and principal) cash flows of a bond with time to maturity $\mathbf{t_x}$

t_k: time from now to the cash flow at point $\mathbf{t_k}$

e: exponential constant

α_1, α_2, α_3, β: parameters to be obtained by **estimation**

Once the parameters α_1, α_2, α_3, β have been estimated we can then use Formula 4-12 to calculate the interest rate at any point t in time as long as $t > 0$ and smaller than or equal to the value of t for the time of the cash flow with the longest time to maturity in the set of bonds that we used to estimate the parameters of α_1, α_2, α_3, β.

Formula 4-12: Nelson & Siegel (1987) parametric yield curve calculation

$$y(t) = \alpha_1 + (\alpha_2 + \alpha_3)(\beta/t)\left(1 - e^{-t/\beta}\right) - \alpha_3 e^{-t/\beta}$$

where:

y(t): interest rate for a cash flow at time t

t: time from now to the cash flow at point **t**

e: exponential constant

α_1, α_2, α_3, β: parameters obtained by estimation using Formula 4-11 and nonlinear least squares estimation

Lab Exercise 4.3 implements both the estimation of the parameters α_1, α_2, α_3, β**:** for Formula 4-11 and the actual calculation of point on the estimated yield curve using estimated values for parameters α_1, α_2, α_3, β as inputs to Formula 4-12 for a range of equally spaced values for t.

Lab Exercise 4.3: Parametric Yield Curve Estimation

1. Open the file **YCurveParametric.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the yield curve calculation shown above.
2. Using the R application load the file **YCurveParametric.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseYCurveParametric()

...

R> RunYCurveParametric()

...

R> YCurveParametricReports()
```

4.1.2.4 Sources of Ready-made Yield Curve Data

After taking a look at the procedures for estimating yield curves and the resulting interest rates we need for valuation it should be clear that this is no trivial task, even in the very simplest of circumstances and taking the simplest possible method. It therefore often pays to obtain yield curve data ready from an external source such as either a commercial data vendor or a specialised calculation service.

4.1.3 Beta and the Cost of Equity

Being able to estimate yield curves will carry us a very long way in practical valuation tasks, but we need some more machinery before we can embark on valuing equity instruments and whole businesses. In order to be able to apply discounted cash flow pricing to equities we need a risk measure for equity analogous to the interest rate applicable in the discount rate for debt instruments.

The most commonly used pricing approach for equities is the Capital Asset Pricing Model (CAPM). The Capital Asset Pricing Model is based on the assumption that the variance of returns is the right measure of risk in the context of pricing, and the only risk that cannot be diversified will be rewarded.

In Formula 4-13 we can see the method for calculating the beta of an instrument relative to a market benchmark. Beta is the undiversifiable risk of an instrument or portfolio relative to the market as measured by a suitable market benchmark. A higher risk and thus higher rewards mean a higher beta, whereas a lower beta means less risk and thus lower expected returns.

Formula 4-13: Beta

$$\beta_{P,B} = Cov(\acute{R}_P, \acute{R}_B) / \sigma_B^2$$

where:

$\beta_{P,B}$: beta of instrument **P** with respect to benchmark **B**

$\acute{R}_P$: time series of returns of the portfolio for periods 1 to **n**

$\acute{R}_B$: time series of returns of the benchmark for periods 1 to **n**

$Cov(\acute{R}_P, \acute{R}_B)$: covariance between returns of the portfolio and the benchmark

σ_B^2: benchmark variance

n: total number of periods under consideration

Once we have calculated the beta for an instrument we can proceed to calculate its **cost of equity**. The cost of equity can be used to vale an equity instrument (stocks or investment fund shares) via the discounted cash flow method. The cost of equity is then used in an analogous manner as the time-dependent interest rate for debt instruments.[1] Formula 4-14 shows how to obtain the cost of equity if we know the beta of an instrument, the risk free rest of interest applicable, as well as the long-term average return of the market index.

Formula 4-14: Cost of equity

$$C_{Equity} + R_{Rf} + \beta_{P,B} \times (E(R_B) - R_{Rf})$$

where:

C_{Equity}: cost of equity for equity instrument

$\beta_{P,B}$: beta of instrument **P** with respect to benchmark **B**

R_{Rf}: risk free interest rate. Usually this is taken to be the interest rate for long-term government bonds at a suitable maturity such as 10, 15, 20, 25, or even 30 years. This we can read of or calculate from a suitable yield curve that represents such credit risk free government bonds.

$E(R_B)$: expected return of the market benchmark. This is simply the long term average for equity returns that we can again estimate from a time series of returns for a market benchmark we plan to use.

[1] See Damodaran (2002) for a detailed discussion of equity valuation methods.

In many valuations we will use the cost equity directly, but sometimes we need to use an average cost of capital instead, such as when we need to value an entire firm rather than just the equity issued by the firm. Formula 4-15 provides a method for calculating the weighted average cost of capital using the cost of equity and debt as well the total values for outstanding equity and debt capital.

Formula 4-15: Weighted average cost of capital (WACC)

$$C_{WACC} = C_{Equity} \cdot \left[E/(E+D)\right] + C_{Debt} \cdot \left[D/E+D\right]$$

where:
C_{WACC}: weighted average cost of capital
C_{Equity}: cost of equity
C_{Debt}: cost of debt
E: total value of equity issued by the firm
D: total value of debt issued by the firm
[E/(E + D)]: proportion of equity issued by the firm to total of all capital
[D/(E + D)]: proportion of debt issued by the firm to total of all capital

Lab Exercise 4.4 implements the calculations for beta, cost of equity, and the weighted average cost of capital using some hypothetical index and instrument return data.

Lab Exercise 4.4: Cost of Equity

1. Open the file **CostOfEquity.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the valuation calculations shown above.
2. Using the R application load the file **CostOfEquity.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseCostOfEquity()

…

R> RunCostOfEquity ()

…

R> CostOfEquityReports()
```

4.1.4 Option Pricing

So far we have looked at discounted cash flow valuation as well as related tools for the estimation of yield curves or the cost of equity, which will allow us to value already calculated prices for a large number of different assets covering both debt obligations and equity rights.

There is however another large set of assets that is derived from debt obligations and equity rights. These are option rights—the right, but not the obligation, to either sell or buy some other asset like a debt obligation or equity right. The most commonly used method in practice is the Black & Scholes option valuation formula, developed by Fischer Black, Myron Scholes and Robert Merton in the early 1970s.[2] This method is set out in Formula 4-16 and illustrated in Lab Exercise 4.5.

Formula 4-16: Black & Scholes option pricing formula

$$\text{Value}_{\text{CallOption}} = \text{spot} * N(d_1) - \text{strike} * N(d_2) * e^{rt}$$

$$\text{Value}_{\text{PutOption}} = -\text{spot} * N(d_1) + \text{strike} * N(d_2) * e^{rt}$$

$$d_1 = \frac{LN\left(\left(\text{spot} * e^{rt}\right)/\text{strike}\right) + \left(\sigma^2 t\right)/2}{\sigma\sqrt{t}}$$

$$d_2 = \frac{LN\left(\left(\text{spot} * e^{rt}\right)/\text{strike}\right) - \left(\sigma^2 t\right)/2}{\sigma\sqrt{t}}$$

where:

t: time to expiry as a fraction of year (365 days)

σ_m: annualised volatility of the underlying

r: continuously compounded risk-free interest rate

N(d): standardised normal cumulative probability function

[2] See Black and Scholes (1973) as well as Merton (1973).

Lab Exercise 4.5: Option Pricing

1. Open the file **SimpleBSOptionVal.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the option valuation calculation introduced above.
2. Using the R application load the file **SimpleBSOptionVal.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseSimpleBSOPtionVal()

...

R> RunSimpleBSOPtionVal ()

...

R> SimpleBSOPtionValReports()
```

4.2 Valuing Financial Assets

Now that we have a basic set of valuation tools let us take a look at the features of some of the most common financial instruments and how our tools introduced so far might help in the valuation of such instruments in a building block manner.

Finally to we will round up the first part of the chapter by taking a look at the most popular building blocks for financial instruments that should be in any valuation toolbox.

4.2.1 Valuing Bonds and Other Debt

Debt instruments are contracts between the issuer and the investor(s) by whom the issuer receives a lump-sum payment from the investor(s) and obliges himself to repay that lump-sum plus a pre-set set of interest (fixed or by formula) according to a pre-defined payment schedule.

4.2.1.1 Zero Bonds and Equivalent Securities and Loans

The simplest kinds of debt instruments are zero coupon bonds and equivalent securities and loans. A zero coupon bond is issued at a deep discount to its par or redemption value and then redeemed in one single payment that constitutes both a return of capital and a component that represents the required interest income for the entire life of the bond for the money originally received by the issuer from the investor when the instrument was issued or the loan taken out.

The cash flows involved can be illustrated in a Cash Flow Timeline diagram. Figure 4.5a shows such a diagram for a zero coupon bond or equivalent security or loan.

Figure 4.5b shows the data model building blocks involved when capturing data in a Zero Coupon Bond.

To value a Zero Coupon Bond or equivalent security or loan you can use the basic formula for calculating the present value of future cash flows by applying an appropriate interest rate for discrete compounding periods such as an annual interest or an interest rate for compounding at six month intervals.

We reproduce the formula in Formula 4-1, introduced earlier.

Formula 4-1: Valuation using annual compounding and an annual interest rate

$$\text{Price}_{\text{CasFlow}} = \text{PV}_{\text{CasFlow}} \frac{F}{(1 + r_{\text{annual}})^{-T}}$$

where:

F: face value of the instrument

T: the time to maturity in number of years

r_{annual}: the annual interest rate applicable

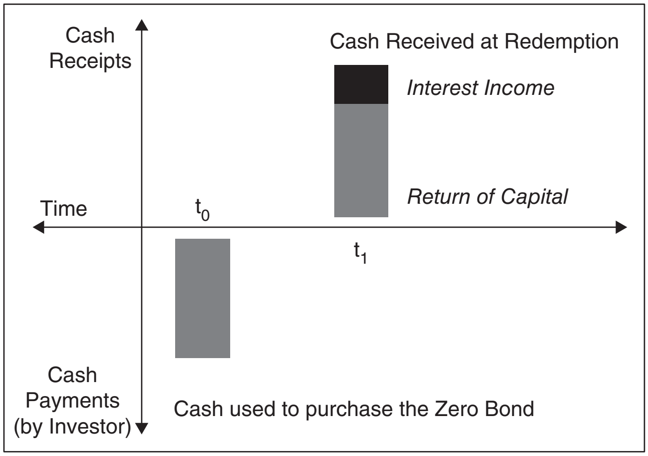


Figure 4.5a: Cash flow timeline for a zero coupon bond or equivalent security or loan.

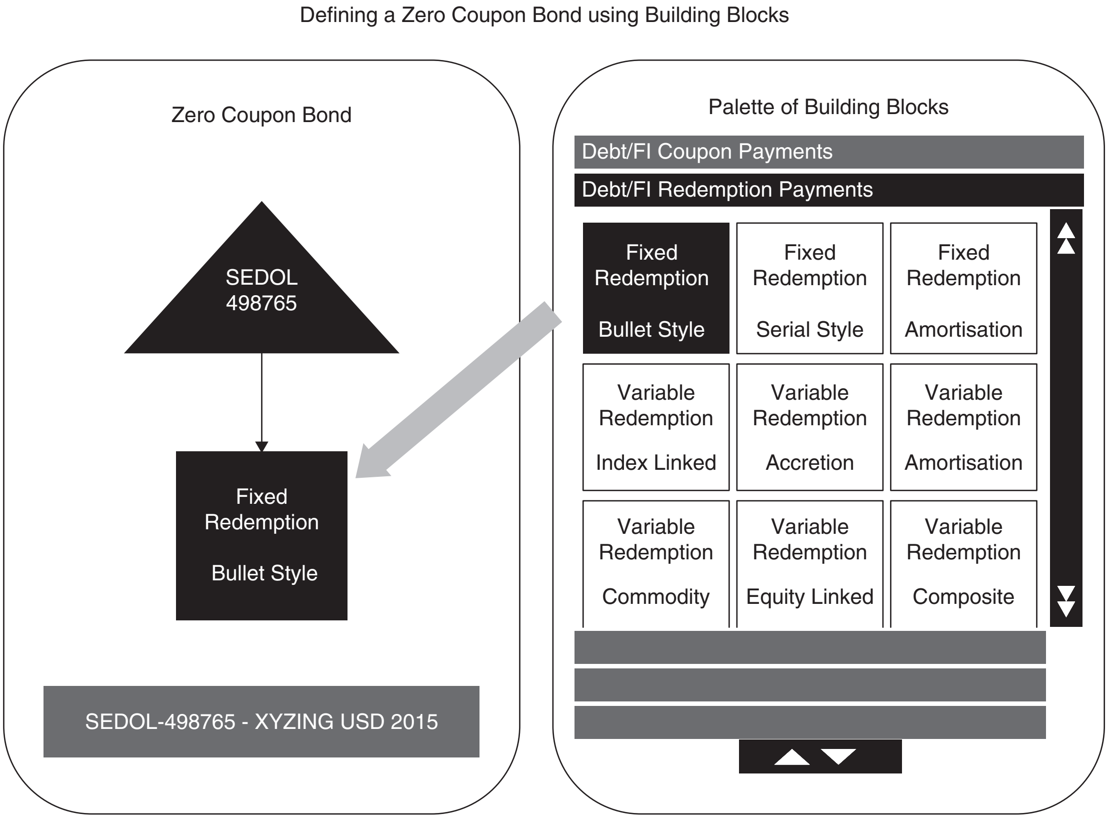


Figure 4.5b: Defining a zero coupon bond.

Formula 4-1 also works for shorter or longer compounding periods. For instance if interest is supposed to be compounded every six months then the rate $\mathbf{r}_{\text{six month}}$ should be used instead of $\mathbf{r}_{\text{annual}}$. Likewise instead of $\mathbf{T}_{\text{in Years}}$ you need to use $\mathbf{T}_{\text{in Half-Years}}$ and for $\mathbf{T}_{\text{in Years}}$ the equivalent six months figures can be calculated as follows: $\mathbf{T}_{\text{in Half-Years}} = 2 * \mathbf{T}_{\text{in Years}}$ and $\mathbf{r}_{\text{six month}} = (1/2) * \mathbf{r}_{\text{annual}} = 0.5 * \mathbf{r}_{\text{annual}}$.

Formulas 4-17 and 4-18 show how **r** can be adapted to compounding periods other than a year in the general case. Use Formula 4-16 to translate annual rate into one for another period and Formula 4-18 to translate the time to maturity from a value measured in years into one measured in other types of periods like months or quarters. After converting both the rate of interest and the number of periods to maturity you can then again use Formula 4-1, but this time substitute $\mathbf{r}_{\text{period}}$ for $\mathbf{r}_{\text{annual}}$ and $\mathbf{T}_{\text{in Periods}}$ for $\mathbf{T}_{\text{in Years}}$.

Formula 4-17: Translating an annual rate into a rate for other compounding periods

$$r_{period} = (1/n) * r_{annual}$$

where:

r_{period}: equivalent of the annual interest rate for the period

r_{annual}: annual interest rate

n: number of the periods equivalent to exactly one year

Formula 4-18: Time to maturity for other compounding periods

$$T_{in\ Periods} = n * T_{in\ Years}$$

where:

$T_{in\ Periods}$: time to maturity of the debt in number of periods

$T_{in\ Years}$: time to maturity of the debt in number of years

n: number of the periods equivalent to exactly one year

Often you may have the interest rate as a rate for continuous compounding rather than for a given compounding period like a year. In this case you will need to use a slightly different formula. We have already introduced this in Formula 4-4, which shows how to calculate the price of a zero bond or equivalent if you have the interest rate used for continuous compounding. For convenience we have repeated Formula 4-4 here again.

Formula 4-4: Valuing a cash flow using a continuous compounding interest rate

$$Price_{CashFlow} = PV_{CashFlow} = \frac{F}{e^{Tr}} = Fe^{-T_r}$$

where:

$PV_{CashFlow}$: present value or price of future cash flows in today's money

F: cash flow amount to be received at time t in the future

T: time to maturity in number of years

r: continuously compounded annual interest rate

e: exponential constant

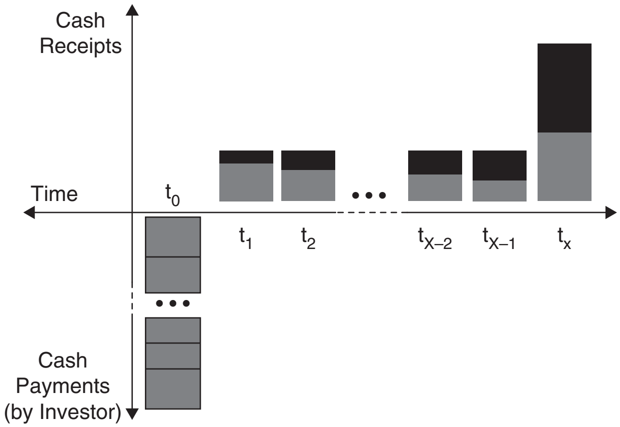


Figure 4.6a: Cash flow timeline for a coupon bond or equivalent security or loan.

4.2.1.2 Fixed Coupon Bonds and Equivalent Securities and Loans

A large number of bonds and some equivalent instruments as coupon paying certificates of deposit have coupon or interest payments at regular intervals throughout the life of the instrument in addition to a redemption payment returning the capital at the end of the life of the instrument.

Figure 4.6a shows the cash flows involved on a cash flow timeline diagram. Looking at this diagram you may come to the conclusion that in fact the instrument is very much like a series of zero bonds—one for each coupon and the redemption—all packaged into a single unit. This in fact is a very useful view since it lets us reuse the pricing formulas we used for zero bonds. The price of the coupon bond in this case is simply the sum of the prices of the individual coupon and the redemption cash flow all added together.

Figure 4.6b and Figure 4.6c show the data model building blocks involved in capturing data on a Fixed Coupon Bond or an equivalent instrument.

If we need a quick shorthand method for calculating the price of a coupon paying bond we can also use Formula 4-5, which we introduced at the beginning of the chapter.

However, as this formula does not allow us to use different interest rates for the cash flows that occur at different points it can only approximate the price. It can be very useful if we need a fast and easy way to calculate an approximate price. For higher accuracy, however, we need to determine the details of each cash flow of the bond, and treating each like a separate Zero Coupon Bond, by calculating a price for each using the appropriate interest rate from a suitable yield curve.

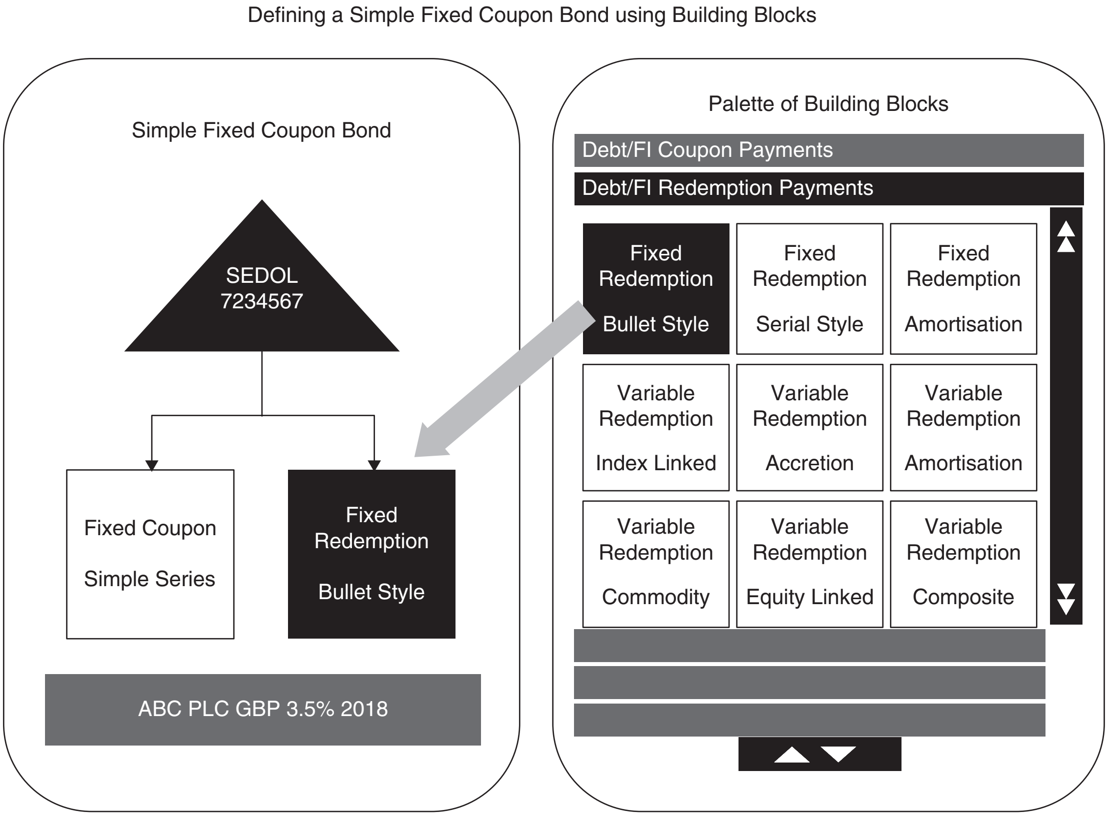


Figure 4.6b: Coupon Bond composition.

Formula 4-5: Valuing a Coupon Bond using a continuous compounding interest rate

$$\text{Price}_{\text{CouponBond}} = \frac{F_{\text{Coupon}}}{e^{r}-1} \times \left[1 - \frac{1}{e^{Nr}}\right] + \frac{F_{\text{Principal}}}{e^{Nr}}$$

where:

N = T × [1/n)

$PV_{CashFlow}$: present value or price of future cash flows in today's money

F_{Coupon}: coupon cash flows from the bond

$F_{Principal}$: cash flows from repayment of the principal of the bond

T: time to maturity in number of years

n: number of coupon payments per year

r: continuously compounded annual interest rate

e: exponential constant

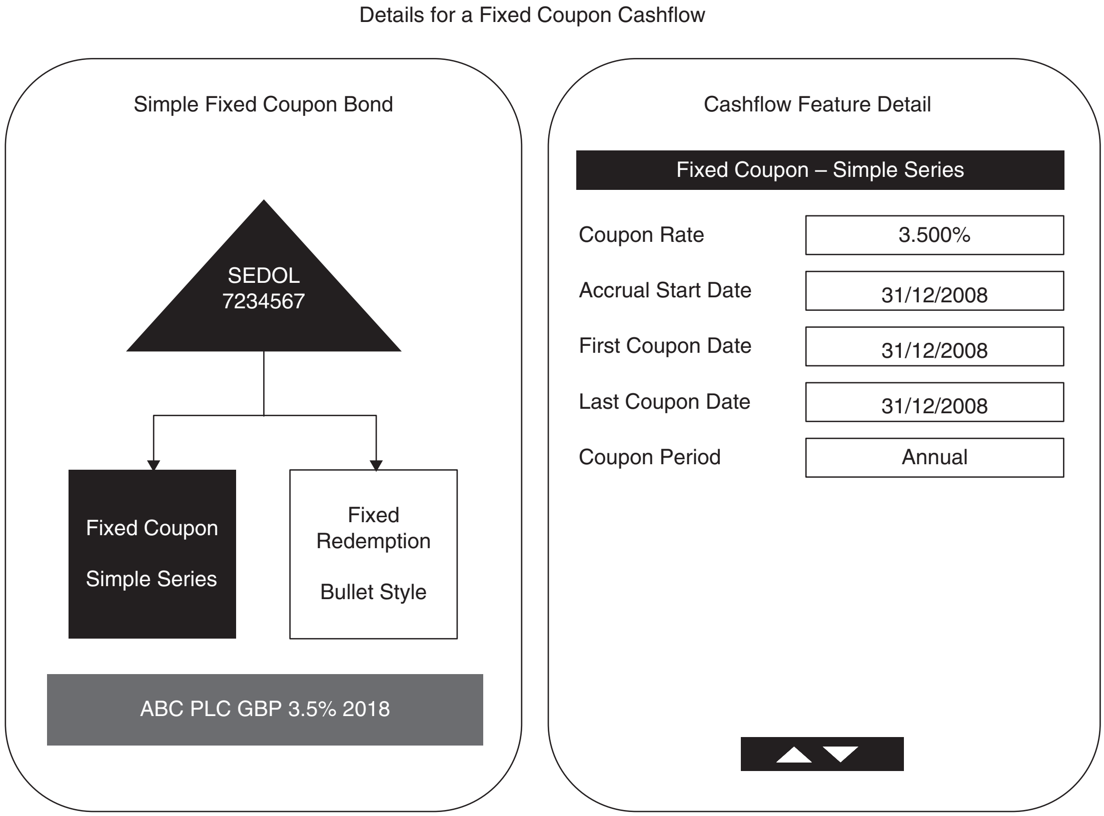


Figure 4.6c: Coupon stream details.

4.2.1.3 Bonds with Floating Coupons and Other Features

Another common form of a bond are Floating Rate Notes. They pay floating or resettable interest rates. In that case instead of paying a given coupon of say X% over the life of the instrument it pays a coupon that is based on some market interest rate like the LIBOR. At periodic intervals the rate is reset using a formula such as LIBOR + Y% and the following coupon or coupons until the next rate reset are then based on this rate as fixed at the last reset date. In the next chapter we will show how the formulas used for pricing Zero Coupon Bonds can again be applied to Floating Rate Notes. In Figure 4.7 you can see an illustration of how a floating rate without a call feature can be represented using the building block method. Figure 4.8 shows a building block representation of a simple zero bond in comparison.

Floating Rate Notes also sometimes have a cap or a floor that defines the maximum or minimum to which the rate can be reset. Here, as we will show later our zero bond formulas are no longer sufficient on their own, and we need to make use of option pricing tools to price the added feature. Again we will look at how to price these features in the following chapter.

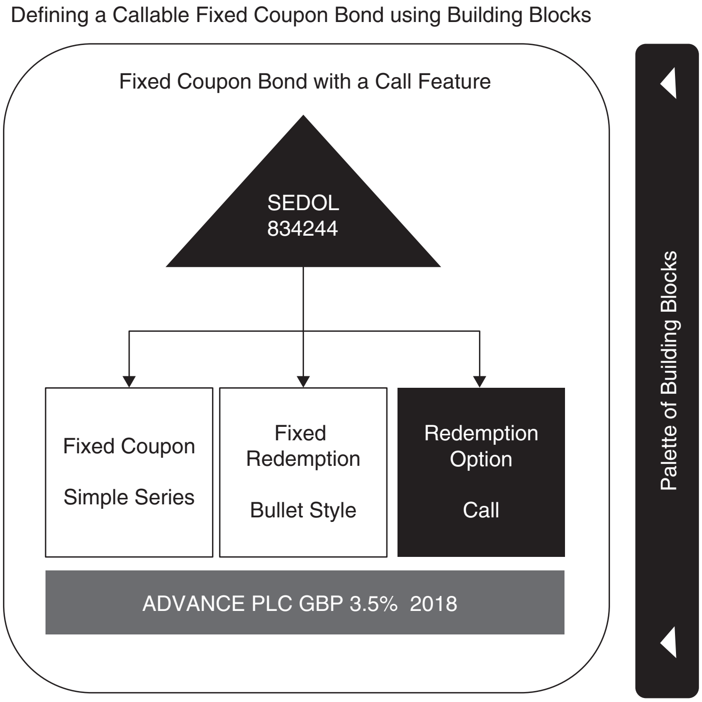


Figure 4.7: Defining a callable Fixed Coupon Bond using building blocks.

Both fixed and floating coupon instruments also often offer the issuer the ability to call; that is to redeem the instrument early; and some bonds allow the investor to put the bonds back to the issuer; that is, they can ask the issuer to redeem their bonds early. Redemption put features for investors and redemption call features for issuers represent options that again could be calculated separately from the value of the rest of the bond and just added to the total price for the instrument.

Other features that can be added to a bond can usually be dealt with in a similar way.

4.2.2 Valuing Equities

Equity shares represent ownership rights in the assets of a specific company. As such, the purchaser of equity shares is entitled to a proportion of the income and capital of the company. Fund units are either equity shares in a fund or entitlements in a fund established as a trust rather than a company. In either case the main difference from equity shares is that the underlying asset pool is not a business but rather a portfolio of financial instruments.

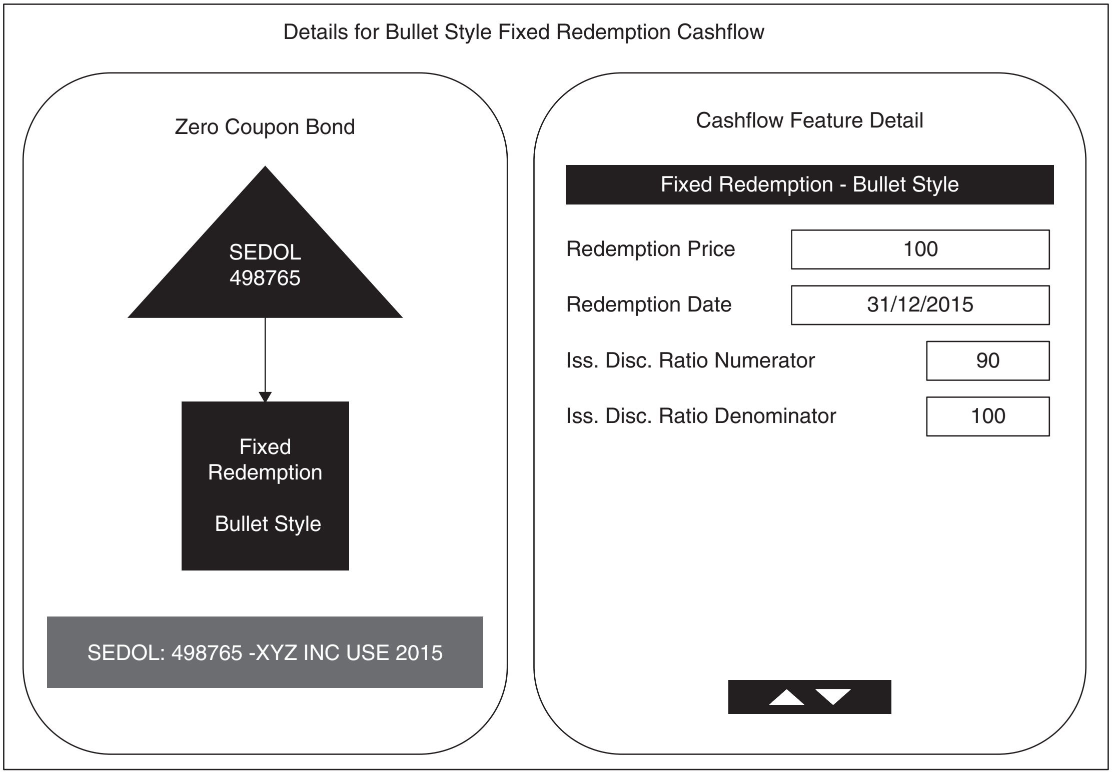


Figure 4.8: Building block view of a Zero Coupon Bond.

See Chapter 3 for a more detailed description of both types of Equity and the related Cash Flows.

4.2.2.1 Valuation Models for Dividend Paying Equities

All equities ultimately derive their value from some form of equitable share of the cash flows generated by one or more underlying assets. Usually, this is a business or in the case of fund units a portfolio of financial assets or real assets like real estate properties. Equities are claims on the residual cash flows from income and capital after all other obligations such as debt interest and repayment and taxation have been met. Equities can thus again be valued with tools derived from those we used for valuing debt.

In a very simple case we might find an equity that has a long track record, paying a steadily growing divided paid for by a business in a very stable industry and very stable economic environment with profits growing at a steady rate that could be maintained indefinitely. The steadily growing dividend cash flows of this scenario are illustrated in Figure 4.10.

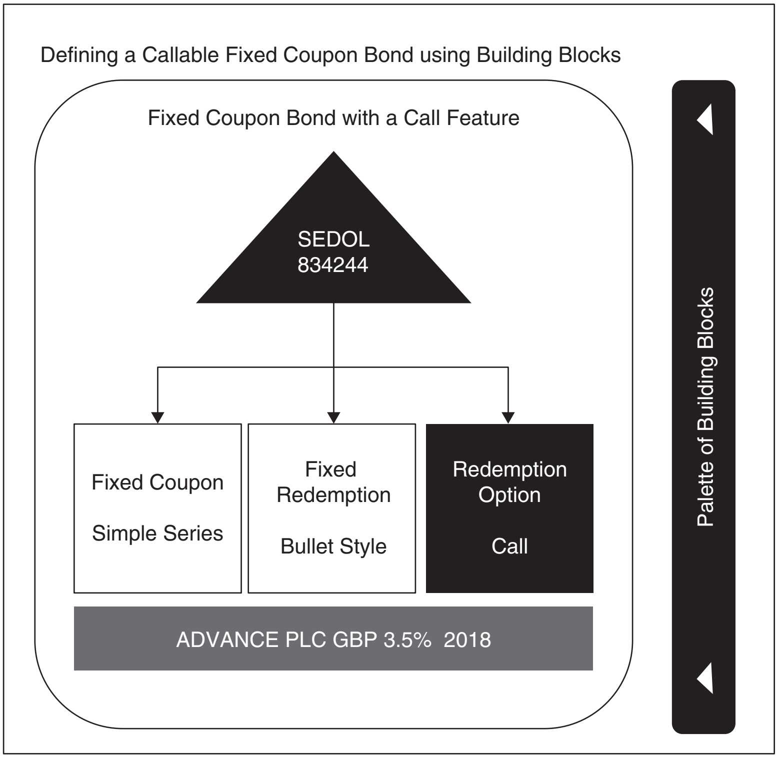


Figure 4.9: Building block view of a callable Fixed Coupon Bond.

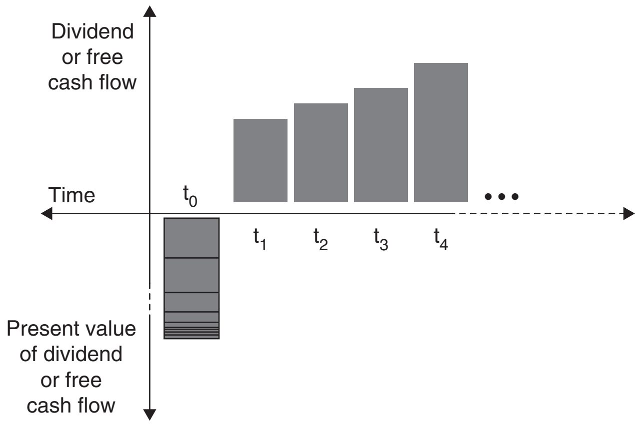


Figure 4.10: Growing dividends or free cash flows used for valuing equity.

In this case we can use the simple dividend discount model for equity valuation shown in Formula 4-19.

Formula 4-19: Simple dividend discount equity valuation

$$V_{Equity} = d_{current} / (C_{Equity} - g_{horizon})$$

where:

V_{Equity}: value of the equity share

$d_{current}$: current dividend paid on the equity share

C_{Equity}: cost of equity for equity instrument, representing the appropriate "interest rate" to discount the dividend stream

$g_{horizon}$: estimated steady sustainable growth of dividend stream

Many times you will find that even for a very simple rough estimate the assumptions of the simple dividend discount equity valuation model are unrealistic.

A very common case in valuation are firms that grow at very fast rate for some years and then are expected to settle down at a rate of growth that is sustainable indefinitely. In this case you need to value the equity by valuing the expected divided for each of the first n years using the simple debt valuation model from Formula 4-1, then calculate a value for the dividends from the year after year n until forever using simple dividend discount equity valuation from Formula 4-19, but using the correct values for year n. Then add all these valuations to a single total value for the equity. This valuation model is shown in Formula 4-20. Figure 4.11 illustrates the cash flow pattern of the two stage growth model.

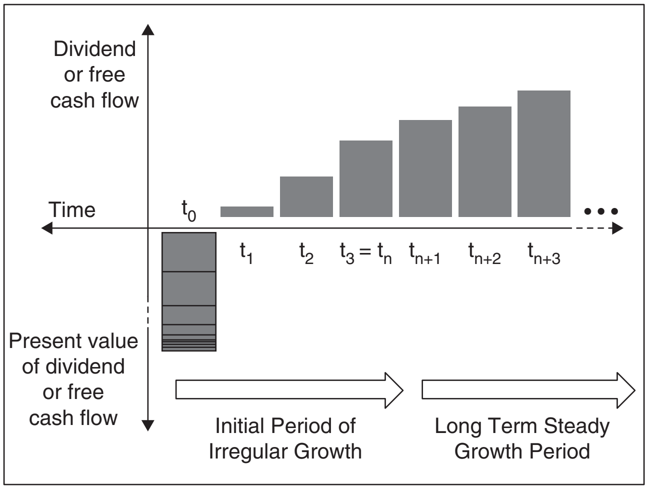


Figure 4.11: Cash flow growth in the two stage growth model.

Formula 4-20: Two stage dividend discount equity valuation

$$\mathbf{V_{Equity}} = \left(\sum_{t=1}^{t=n} \mathbf{d_t} \big/ \left(1 + \mathbf{C_{Equity}}\right)^{t}\right) + \mathbf{V_{longterm}} \big/ \left(1 + \mathbf{C_{Equity}}\right)^{n}$$

where:

$\mathbf{V_{Equity}}$: value of the equity share

$\mathbf{d_t}$: dividend paid on the equity share in year t

t: index of year in the first 1 to t years (i.e., 1 in first year, 2 in second year, etc., and n in the final year where t = n)

$\mathbf{C_{Equity}}$: cost of equity for equity instrument, representing the appropriate "interest rate" to discount the dividend stream

$\mathbf{g_{longterm}}$: estimated steady sustainable growth of dividend stream from year

$\mathbf{d_{n+1}}$: expected dividend on the equity share in the year after year t

$\mathbf{V_{longterm}}$: $= \mathbf{d_{n+1}/(r - g_{longterm})}$

If the dividend growth is the same for the first n year you can take a slight short-cut using the model shown in Formula 4-21.

Formula 4-21: Simplified two stage dividend discount equity valuation

$$\mathbf{V_{Equity}} = \mathbf{V_{IntialPeriod}} + \mathbf{V_{LongTerm}}$$

$$\mathbf{V_{InitialPeriod}} = \left(\mathbf{d_{current}} * \left(1 + \mathbf{g_{initial}}\right) * \left(1 - \left(\left(1 + \mathbf{g_{initial}}\right)^{n} \big/ \left(1 + \mathbf{C_{Equity}}\right)^{n}\right)\right)\right) \Big/ \left(\mathbf{C_{Equity}} - \mathbf{g_{initial}}\right)$$

$$\mathbf{V_{LongTerm}} = \mathbf{d_{n+1}} \Big/ \left(\left(\mathbf{C_{Equity}} - \mathbf{g_{longterm}}\right) * \left(1 + \mathbf{C_{Equity}}\right)^{n}\right)$$

where:

$\mathbf{V_{Equity}}$: value of the equity share

$\mathbf{V_{InitialPeriod}}$: equity valuation for the first n years

$\mathbf{V_{LongTerm}}$: equity valuation for the period after year n

$\mathbf{d_{current}}$: expected dividend paid on the equity at the end of year 1

$\mathbf{g_{initial}}$: estimated growth for the first n years

$\mathbf{g_{longterm}}$: estimated steady sustainable growth of dividend stream after year n

n: number of years initial period: e.g., 3 or 5 years

$\mathbf{C_{Equity}}$: cost of equity for equity instrument, representing the appropriate "interest rate" to discount the dividend stream

$\mathbf{d_{n+1}}$: expected dividend on the equity share in the year after year n

The model in Formula 4-20 is the most common in practical use since more accurate valuations can often be obtained from detailed forecasts that can be made for a certain number of years, whereas the value of cash flows from years beyond the forecast period is estimated using the simple steady state growth equity valuation model.

4.2.2.2 Valuation Models for Equities Based on Free Cash Flows

Up to now we have taken the basis for equity valuation to be the dividend stream they pay. In practice, however, you will find many equities do not pay a regular dividend. This is often the case with younger companies experiencing strong growth. In this case we can still value the equity but need to use a replacement for the dividend cash flows. This replacement is a value called **Free Cash Flow to Equity**, which is the net cash flow for the company in each period after paying all other claims like operating and other cash-based expenses, interest, and principal repayments on debt due as well as taxes.

We will look at how to calculate and value free cash flows in more detail in the following chapter.

4.2.3 Valuing Forwards and Futures

Forwards are simply trades concluded in advance where the exchange of cash against delivery of the asset takes place at some time in the future. A Futures Contract is a highly standardised alternative form of a forward that can be traded on an exchange because of its standardisation.

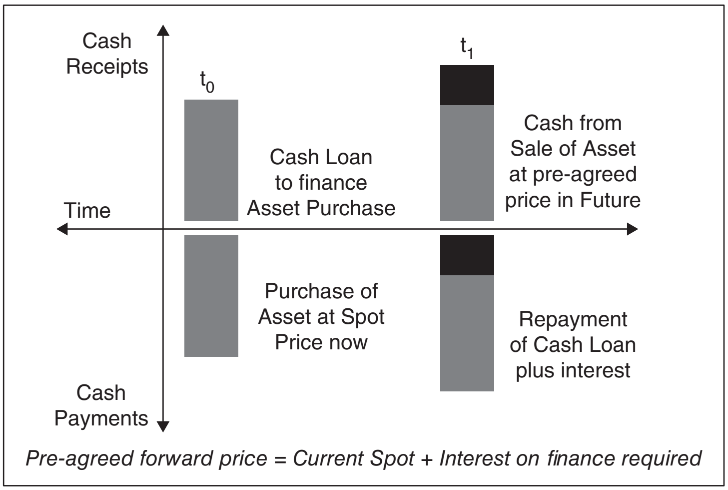


Figure 4.12: Cash flow pattern of a forward agreement.

4.2.3.1 Valuing Forwards and Futures of Financial Assets without Income

Prices for futures and forwards are not necessarily exactly equal because of the difference in actual cash flow patterns between the two. In practice, however, this difference is usually ignored and a single formula is used for both. The basis for valuing both forwards and futures is the fact that they are essentially debt instruments denominated in terms of a certain quantity of an asset. In Figure 4.7 you can see an illustration of how a floating rate not with a call feature can be represented using the building block method. Figure 4.8 shows a building block representation of a simple zero bond in comparison.

To illustrate this consider the following alternatives for both the buyer and the seller of the forward or future:

The buyer of a forward could borrow money and buy the asset in the spot market. If the seller did not want to offer a forward, he or she could just sell the asset in the spot market and invest the proceeds in the money market, earning the given market interest rate for deposits or short-term instruments. Figure 4.13 illustrates the building block structure of a simple forward contract without any income and cost features.

Formula 4-22, which is the model for valuing futures and forwards on assets not paying any income and not having any storage cost, thus is based on the value of the asset in the spot market and the interest that could be earned on the equivalent cash amount in the money market at the risk-free rate.

Formula 4-22: Value of a Forward or Future of a security without income

$$V_{Forward} = V_{Spot} * (1 + r)^T$$

where:

$V_{Forward}$: the theoretical value of the forward or future contract

V_{Spot}: value of the some position in the instrument in the spot market

T: time to maturity of the contract of years

r: market interest rate appropriate for discounting the dividend stream

4.2.3.2 Valuing Forwards and Futures of Financial Assets with Income or Storage Costs

Some assets (e.g., certain equities) actually pay a known or expected income over the period of the forward or future. Other assets such as oil or other commodities will require storage at a known cost. In both cases we will need to extend our valuation model to account for

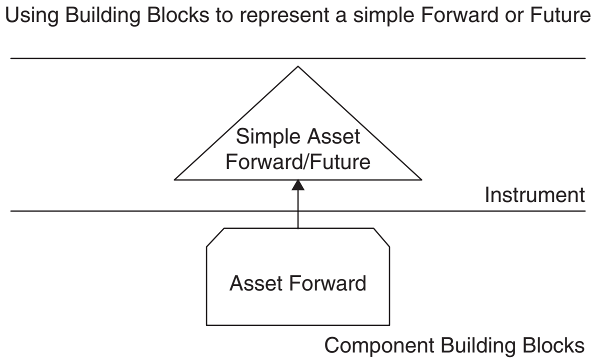


Figure 4.13: Building block view of a simple asset-based forward or futures contract.

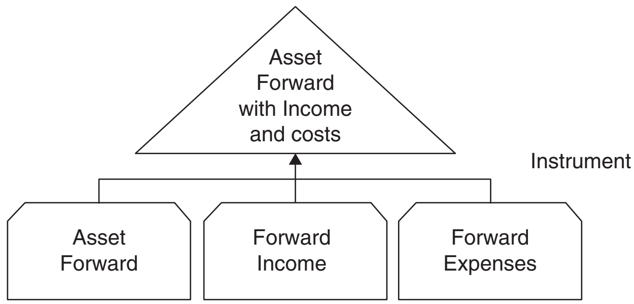


Figure 4.14: Building block view of a forward contract with income or storage costs.

the cost. If the asset receives some income between the start of the forward and the time delivery takes place then we need to subtract the appropriately discounted income from the plain forward price since the income reduces the funding costs for the provider of the forward. If the asset requires the holder to pay expenses for storage, the appropriately discounted expenses need to be added to the plain forward price since the expenses add to the funding costs for the provider of the forward. Formula 4-23 shows the extended forward valuation model that includes both an adjustment for income received and expenses paid. Figure 4.14 illustrates the building block structure of a forward contract with income and costs.

Formula 4-23: Value of a Forward or Future of a security with income and/or costs

$$V_{Forward} = V_{Spot} * (1 + r)^T - \sum_{t=1}^{t=n} \left(d_t / (1 + r_d)^t \right) + \sum_{t=1}^{t=m} \left(e / (1 + r_e)^t \right)$$

where:

$V_{Forward}$: theoretical value of the forward or future contract

V_{Spot}: value of the same position in the instrument in the spot market

T: time to maturity of the contract of years

r: market interest rate assuming annual compounding

$r_{d:}$ **r** * period_between_income_payments_in_year_fractions

$r_{e:}$ **r** * period_between_expense_payments_in_year_fractions

d_t: income from asset in period t for any t in between 1 ... n

e_t: income from asset in period t for any t in between 1 ... m

t: time index terms of no periods for either income (d_t) or expenses (e_t)

n: number of income payments from start to end of the contract

m: number of expense payments from start to end of the contract

4.2.4 Valuing Options

If we wanted to value a call option, that is the right but not obligation to buy some given equity shares (e.g., shares for Marks & Spencer Plc, Daimler AG or General Motors) at a pre-agreed price at some future date we can use our Black & Scholes option valuation tool from Formula 4-16, introduced earlier in the chapter.

Formula 4-16: Black & Scholes option pricing formula

$$Value_{CallOption} = spot * N(d_1) - strike * N(d_2) * e^{rt}$$

$$Value_{PutOption} = -spot * N(d_1) + strike * N(d_2) * e^{rt}$$

$$d_1 = \frac{LN\left((spot * e^{rt}) / strike\right) + (\sigma^2 t)/2}{\sigma\sqrt{t}}$$

$$d_2 = \frac{LN\left((spot * e^{rt}) / strike\right) - (\sigma^2 t)/2}{\sigma\sqrt{t}}$$

where:

t: time to expiry as a fraction of year (365 days)

σ: annualised volatility of the underlying

r: continuously compounded interest rate

N(d): standardised normal cumulative probability function

In order to get accustomed to the various valuation methods presented so far, please work through the following exercise on the companion web site of the book.

Lab Exercise 4.6: Financial Instrument Valuation Examples

1. Open the file **FinInstValutions.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the valuation calculations shown above.
2. Using the R application load the file **FinInstValutions.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseFinInstValuations()

...

R> RunFinInstValuations ()

...

R> FinInstValuationsReports()
```

4.3 Valuing Real Assets

Not all assets that we may have in a portfolio will be financial instruments like tradable securities, derivatives or over-the-counter (OTC) derivatives contracts of account-based loan or deposit-based products. Sometimes we may well need to value such entities as an entire business, a real estate property, or some large project asset like a new power utility or a very large innovative new ship. Although each of those are clearly different from financial instruments we can again use our tools to price them.

4.3.1 Valuing a Business

To value a business we can again use our equity valuation tools introduced earlier. Formula 4-24 is an adaptation of the model from Formula 4-20. In this case however we will have to use the weighted average cost of capital.

Formula 4-24: Two-stage cash flow based business valuation

$$V_{Business} = \left(\sum_{t=1}^{t=n} d_t/(1+r)^t\right) + V_{longterm}/(1+r)^n$$

where:

$V_{Business}$: value of the business

d_t: cash flows for the business in year t

t: index of year in the first 1 to t years (i.e., 1 in first year, 2 in second year, etc., and n in the final year where t = n)

r: market interest rate appropriate for discounting the stream of expected cash flows from the business

$g_{longterm}$: estimated steady sustainable growth of the stream of cash flows

d_{n+1}: expected cash flows from the business in the year after year t

$V_{longterm}$: = $d_{n+1}/(r - g_{longterm})$

Formula 4-15, introduced earlier, provides a method for calculating the weighted average cost of capital using the cost of equity and debt as well the total values for outstanding equity and debt capital. We need to use this formula to calculate the correct cost of capital to use for discounting, taking into account any debt funding the business may use in addition to the equity that may have been invested.

Formula 4-15: Weighted average cost of capital (WACC)

$$C_{WACC} = C_{Equity} \cdot [E/(E+D)] + C_{Debt} \cdot [D/(E+D)]$$

where:

C_{WACC}: weighted average cost of capital

C_{Equity}: cost of equity

C_{Debt}: cost of debt

E: total value of equity issued by the firm

D: total value of debt issued by the firm

[E/(E + D)]: proportion of equity issued by the firm to total of all capital

[D/(E + D)]: proportion of debt issued by the firm to total of all capital

4.3.2 Valuing Real Estate Property

In cases where we want to value some real estate property, we can again use the method we just introduced for the valuation of businesses. Some more complex properties may have additional complex features such as options to sell or to buy more units from a multiunit development. Instead of needing a completely new valuation approach though, we can again just value the additional features using other tools such as option valuation and then add the different partial valuations together.

4.3.3 Valuing Large Projects: Ships, Utilities, and More

Sometimes, such as in the case of certain utilities projects of new innovative large ships to be built, it may be better not to use a tool like the business valuation method introduced in Formula 4-24. Instead it may be better to use an alternative, more sophisticated option valuation approach that allows us to capture the value from the optionality such projects entail better than a business/equity valuation approach. In the case of an innovative large new ship, the building and commissioning process may take some years while the future revenues, which because of this time lag are still in the relatively distant future, will depend very heavily on the evolution of markets in the future. The ship building project can thus be seen as a call option on the future business that may result from owning such a ship. Such real options are a good illustration that selecting the right valuation model and thus an appropriate way of representing an instrument's financial characteristics should be done with great care. In some cases you even have several such models in parallel for an instrument so that you can compare them.

Before concluding this chapter you may find it useful to work through Lab Exercise 4.7, which will allow you to explore some of the aspects of the implementation of the previous approaches for valuing real assets in more detail.

Lab Exercise 4.7: Real Asset Valuation Examples

1. Open the file **RealAssetValutions.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the valuation calculations shown above.

2. Using the R application load the file **RealAssetValutions.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseRealAssetValuations()

…

R> RunRealAssetValuations ()

…

R> RealAssetReports()
```

CHAPTER 5

Implementing Valuation Models

We outline and demonstrate the four main steps to implement and refine valuation models in an evolutionary iterative way. We show how to build and use checklists to guide your valuation model implementation, and how to master your valuation model meta-, input, and output data within the framework of the data model blueprint introduced earlier.

5.1 Valuation Model Implementation—Step by Step

There are four main steps in a valuation model implementation using the building block approach (Figure 5.1).

First, you will need to decide how the instruments in your domain are to be decomposed into atomic building blocks. This can be as simple as adopting a standard palette of atomic building blocks or may involve the development of a specialised palette of building blocks yourself if needed.

Second, you will need to create the valuation models themselves and then map each building block to at least one of your valuation models.

Third, you will need to build transformation models, identify some suitable source, or provide your own estimates for each of the inputs that need to be transformed before they can be used in your valuation models. Typical examples of such inputs are market yield curves, beta values for companies, or equity instruments and volatilities of the prices of instruments used as underlying in an option.

The fourth step is to create an overall process that runs your valuation models, feeds them with data, and writes any results back into the database so that they can be used later from online programs, in reports or in data feeds.

In all but the most trivial cases you would start with only a very small set of valuation models or even just one, and then go on to refine your implementation and add additional models and input data once you have the basic process running to your satisfaction.

We will now look at each step in turn and then briefly consider how best to shape the gradual refinement of your model implementation.

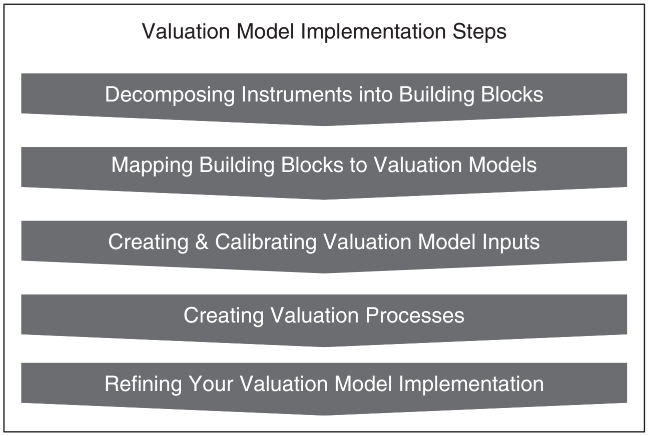


Figure 5.1: Valuation model implementation steps.

5.1.1 Decomposing Instruments into Building Blocks

The first step in a pricing model implementation is to ensure that you can map all required reference data that drives the models into your instrument database. With the building block method this task can be split into two sub steps:

(1) Mapping each atomic building block component for instruments into the database and

(2) Ensuring that you can decompose each instrument into a set of atomic building block components.

In Figure 5.2 you can see an illustration of the second sub step—the process of decomposing an instrument into atomic building blocks. By now you are most likely already familiar with this process but you may find it useful to try Lab Exercise 5.1 on page 119 to get some additional practice.

As for the first sub step—mapping atomic building blocks to the database—you already have seen some examples in Chapter 3. The key to making the building block approach work in practice is to have a database structure that lets you represent all the different building blocks easily and without changes to the database structure.

There are potentially many different ways in which this can be done. In Chapters 2 and 3 we propose an approach based on two structures called Cash Flow Element and Cash Flow Schedule. In Figures 5.3 and 5.4 you can see how the terms from a Fixed Coupon can be mapped into the attributes of a Cash Flow Element and Cash Flow Schedule instance, respectively. In an application that lets you edit the attributes for particular building blocks, any unmapped attributes should be hidden from view. Those attributes that are mapped should be labelled with the accepted market term such as **Coupon Rate** rather than the attribute name of the Cash Flow Element or Cash Flow Schedule to which it is mapped.

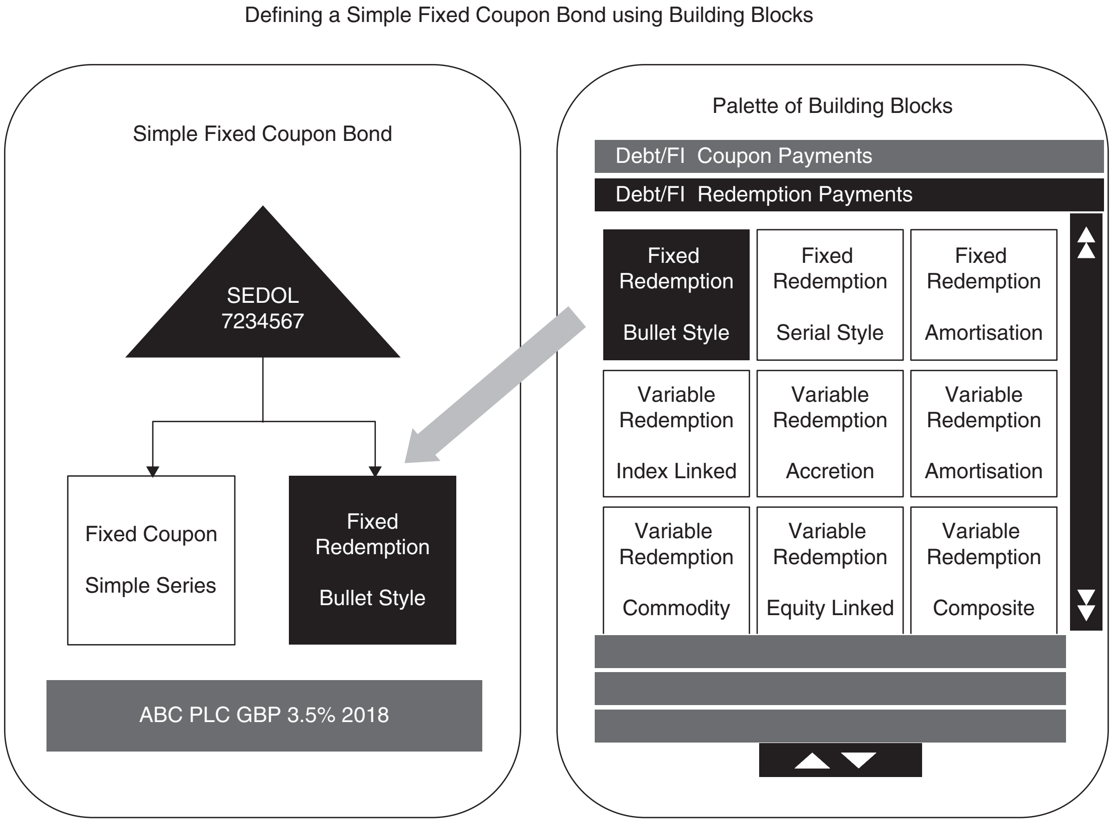


Figure 5.2: Decomposing a fixed coupon bond into building blocks.

Thus shown for a Fixed Coupon building block you would show Coupon Currency, Coupon Rate, First Coupon Date, rather than Base Instrument Identifier, Schedule Start Date, Rate Constant, and so on.

The mapping of a building block to a Cash Flow Element and a set of linked Cash Flow Schedules represents a template that guides you and those who need to provide you with reference data for your modelling effort in where to put each term from the features of a given instrument that can be represented by a particular building block.

If you build your own instrument database, you need to define a mapping for each building block you want to use. Whenever you can though, you should use standardised mappings. This will save you time and make it easier for you to exchange data with other people who also use the building block approach. You can find more information on how to obtain such standardised mappings on the companion web site for this book (http://modelbook.bancstreet.com/).

Cash Flow Element Mapping Pattern for a Fixed Coupon			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Tag	VARCHAR2	255	
Element Type	**VARCHAR2**	**32**	**"Fixed-Coupon"**
Element Analytic Model	VARCHAR2	32	
Element Sequence	NUMBER	22	
Element Operator	VARCHAR2	32	
Element Tag	**VARCHAR2**	**255**	***Coupon Reference***
Element Parent Tag	VARCHAR2	255	
Period Start Date	**DATE**	**7**	***Accrual Start Date***
Period End Date	**DATE**	**7**	***Last Coupon Date***
Is Detachable	NUMBER	22	
Base Instrument Identifier Type	**VARCHAR2**	**32**	**"ISO-Currency-Code"**
Base Instrument Identifier	**VARCHAR2**	**32**	***Redemption Currency***
Base Value	NUMBER	22	
Base Value Denominator	NUMBER	22	
Base Unit	VARCHAR2	32	
Trade Instrument	NUMBER		
Trade Instrument Identifier Type	VARCHAR2	32	
Trade Instrument Identifier	VARCHAR2	32	
Trade Value	NUMBER	22	
Trade Unit	VARCHAR2	32	
Rate Constant	**NUMBER**	**22**	***Coupon Rate***
Rate Factor	NUMBER	22	
Cap	NUMBER	22	
Floor	NUMBER	22	
Minimum Step	NUMBER	22	
Maximum Step	NUMBER	22	
Amortisation Factor	NUMBER	22	
Payout Factor	NUMBER	22	
Is Accumulating	NUMBER	22	
Underlying Instrument Identifier Type	VARCHAR2	32	
Underlying Instrument Identifier	VARCHAR2	32	
Underlying Type	VARCHAR2	32	
Underlying Price Type	VARCHAR2	32	
Index Factor	NUMBER	22	
Index Base	NUMBER	22	
Upper Bound	NUMBER	22	
Lower Bound	NUMBER	22	
Exercise Style	VARCHAR2	32	
Mandatory Exercise	NUMBER	22	
Date Accrual Style	VARCHAR2	32	
Date Accrual Rule	**VARCHAR2**	**32**	***Accrual Rule***
Settlement Style	VARCHAR2	32	
Instrument Build	VARCHAR2	32	
Maturity Period Amount	NUMBER	22	
Protected Obligation	VARCHAR2	32	
Credit Protection	VARCHAR2	32	
Right Owner	VARCHAR2	32	
Paying Party	NUMBER	22	

Figure 5.3: Mapping a fixed coupon to the standardised Cash Flow Element structure.

Cashflow Schedule Mapping Pattern for a Fixed Coupon			
CASHFLOW SCHEDULE ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Cash Flow Schedule Type	**VARCHAR2**	**32**	***Coupon Accrual Schedule***
Sequence	NUMBER	22	
Element Tag	**VARCHAR2**	**255**	***Coupon Reference***
Repetition Period Count	NUMBER	22	
Repetition Period Type	**VARCHAR2**	**32**	***Months***
Repetition Period Length	**NUMBER**	**22**	***6***
Is Adjusted	NUMBER	22	
Schedule Start Date	**DATE**	**7**	***Accrual Start Date***
Schedule End Date	**DATE**	**7**	***Last Coupon Date***
Repetitions	NUMBER	22	
Day Offset Rule	VARCHAR2	32	
Start Offset	NUMBER	22	
Start Offset Type	VARCHAR2	32	
Ex Date Offset Rule	VARCHAR2	32	
Calendar Group Tag	VARCHAR2	255	
Trigger Event Type	VARCHAR2	32	
Grace Period	NUMBER	22	
Grace Period Type	VARCHAR2	32	
Event Threshold Amount	NUMBER	22	
Event Threshold Instrument	NUMBER		
Event Threshold Instrument Identifier Type	VARCHAR2	32	
Event Threshold Instrument Identifier	VARCHAR2	32	
Public Source Type	VARCHAR2	32	
Public Source Count	NUMBER	22	
Notifying Party Group Tag	VARCHAR2	32	
Fixing Party Group Tag	VARCHAR2	32	

Figure 5.4: Mapping a Fixed Coupon to the standardised Cash Flow Schedule Structure.

Lab Exercise 5.1: Decomposing Financial Instruments

1. Obtain a trial account for the Fincore Financial Investment Portal if you have not already done so. You can find details of how to do so on the companion web site (http://modelbook.bancstreet.com/).
2. Download the tutorial notes **LAB_5_1_DecomposingInstruments.pdf** from the companion web site and follow the instructions to compose and decompose a few of the most common financial instruments.

5.1.2 Mapping Building Blocks to Valuation Models

The second of the four main steps in a valuation model implementation is to map the information in your instrument database to your pricing models. In Figure 5.5 you can see how the parameters from the discounted cash flow pricing formula for continuous interest rates, which we introduced in Chapter 4, are mapped to the terms of a Bullet Redemption building block (see Figure 5.5) and other information in the database.

In this example the **Cash flow amount to be received at time t in the future** can be taken directly from the **redemption price** term for a bullet redemption.

The **time to maturity in number of years** has to be calculated as the **number of years between valuation date and redemption date**, where the valuation date is a parameter you normally would provide when running the model; the redemption date can again be taken from the bullet redemption terms.

The **continuously compounded annual interest rate** finally does not come from the terms for a bullet redemption but rather must be read off a suitable yield curve. In the building block approach yield curves are also represented as instruments. The actual curve values for a yield curve can be stored in a different way, either as input parameters for a curve model or as interpolation points. If the curve is stored as interpolation points you would normally find them in the Instrument Analytics table, which we introduced earlier in the book in Chapter 2.

In Figure 5.6 you can see a similar mapping that shows how to obtain the parameters for the Black & Scholes Option Pricing formula shown in Chapter 4 from the terms of an Equity Call Option building block such as the one given in Figure 5.10. In Figures 5.7 and 5.8 you can see the representation of the data for a bullet redemption and floating rate note, respectively. Figure 5.9 provides a representation of the relevant terms for an equity option.

Again, some parameter such as the **Strike Price** can be read directly from the terms of the Equity Call Option building block whereas others, such as the Volatility of the Underlying or the current Spot Price, can be looked up in other parts of the database with the help of

Pricing Model Mapping for a Bullet Redemption	
Model Parameter	**Source**
F: Cash flow amount to be received at time t in the future	***Redemption price in the bullet redemption***
T: Time to maturity in number of years	***Number of years between valuation date and redemption date***
R: Continuously compounded annual interest rate	***Read this from a suitable yield curve***

Figure 5.5: Mapping a bullet redemption to a discounted cash flow pricing model.

Pricing Model Mapping for an Equity Call Option	
Model Parameter	**Source**
T = Time to expiry as a fraction of year (365 days)	***Number of years between valuation date and redemption date***
σ = Annualised volatility of the underlying	***Obtain from Instrument Analytics table***
r = Continuously compounded interest rate	***Read this from a suitable yield curve***
Strike Price	***Strike Price term***
Current Spot Price	***Obtain from Instrument Price table***

Figure 5.6: Mapping an Equity Call option to the Black & Scholes option pricing model.

the terms of the Equity Call Option building block. A few remaining ones like the riskless interest rate and the valuation date have to be set up when the model is configured for a particular purpose or given to the model when it is actually run.

Lab Exercise 5.2: Exploring Model Mappings

1. Download the tutorial notes **LAB_5_2_ExploringModelMappings.pdf** from the companion web site (http://modelbook.bancstreet.com/) to see and explore further model mappings like the ones shown in this chapter but for other common building blocks like equities, floating rate coupons, and index linked redemptions.

5.1.3 Creating and Calibrating Valuation Model Inputs

Although some model inputs—like the coupon rate of a fixed coupon bond or the strike price of an option—are static reference data and can be obtained from the terms of instrument building blocks like coupons, redemptions, and options, all models also need further inputs related either to the instrument itself, its issuer, or the market more generally. The most common one is the yield curve from which to obtain appropriate interest rates to discount the cash flows of the instrument for valuation purposes and option pricing models. Others include the beta for a given firm or equity instrument or the volatility of a share used as underlying in an equity option.

It makes sense to distinguish these inputs into two categories:

First the calculation of constructs like yield curves, which can be called model calibration or calibration for short since a yield curve and similar constructs provide generic parameters for models not directly related to an instrument to be valued.

Cash Flow Element Mapping Pattern for a Bullet Redemption			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Tag	VARCHAR2	255	
Element Type	**VARCHAR2**	**32**	**"Fixed-Redemption-Bullet-Style"**
Element Analytic Model	VARCHAR2	32	
Element Sequence	NUMBER	22	
Element Operator	VARCHAR2	32	
Element Tag	**VARCHAR2**	**255**	***Redemption Reference***
Element Parent Tag	VARCHAR2	255	
Period Start Date	**DATE**	**7**	***Issue or Accrual Start Date***
Period End Date	**DATE**	**7**	***Redemption Date***
Is Detachable	NUMBER	22	
Base Instrument Identifier Type	**VARCHAR2**	**32**	**"ISO-Currency-Code"**
Base Instrument Identifier	**VARCHAR2**	**32**	***Redemption Currency***
Base Value	NUMBER	22	
Base Value Denominator	NUMBER	22	
Base Unit	VARCHAR2	32	
Trade Instrument	NUMBER		
Trade Instrument Identifier Type	VARCHAR2	32	
Trade Instrument Identifier	VARCHAR2	32	
Trade Value	NUMBER	22	
Trade Unit	VARCHAR2	32	
Rate Constant	**NUMBER**	**22**	***Redemption Price***
Rate Factor	NUMBER	22	
Cap	NUMBER	22	
Floor	NUMBER	22	
Minimum Step	NUMBER	22	
Maximum Step	NUMBER	22	
Amortisation Factor	NUMBER	22	
Payout Factor	NUMBER	22	
Is Accumulating	NUMBER	22	
Underlying Instrument	NUMBER		
Underlying Instrument Identifier Type	VARCHAR2	32	
Underlying Instrument Identifier	VARCHAR2	32	
Underlying Type	VARCHAR2	32	
Underlying Price Type	VARCHAR2	32	
Index Factor	NUMBER	22	
Index Base	NUMBER	22	
Upper Bound	NUMBER	22	
Lower Bound	NUMBER	22	
Exercise Style	VARCHAR2	32	
Mandatory Exercise	NUMBER	22	
Date Accrual Style	VARCHAR2	32	
Date Accrual Rule	**VARCHAR2**	**32**	***Accrual Rule***
Settlement Style	VARCHAR2	32	
Instrument Build	VARCHAR2	32	
Maturity Period Amount	NUMBER	22	
Protected Obligation	VARCHAR2	32	
Credit Protection	VARCHAR2	32	
Right Owner	VARCHAR2	32	
Paying Party	NUMBER	22	

Figure 5.7: Mapping a bullet redemption to the Cash Flow Element structure.

Cash Flow Element Mapping Pattern for a Floating Coupon			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Tag	VARCHAR2	255	
Element Type	**VARCHAR2**	**32**	**"Floating-Rate-Coupon"**
Element Analytic Model	VARCHAR2	32	
Element Sequence	NUMBER	22	
Element Operator	VARCHAR2	32	
Element Tag	**VARCHAR2**	**255**	***Coupon Reference***
Element Parent Tag	VARCHAR2	255	
Period Start Date	**DATE**	**7**	***Accrual Start Date***
Period End Date	**DATE**	**7**	***Last Coupon Date***
Is Detachable	NUMBER	22	
Base Instrument Identifier Type	**VARCHAR2**	**32**	**"ISO-Currency-Code"**
Base Instrument Identifier	**VARCHAR2**	**32**	***Redemption Currency***
Base Value	NUMBER	22	
Base Value Denominator	NUMBER	22	
Base Unit	VARCHAR2	32	
Trade Instrument	NUMBER		
Trade Instrument Identifier Type	VARCHAR2	32	
Trade Instrument Identifier	VARCHAR2	32	
Trade Value	NUMBER	22	
Trade Unit	VARCHAR2	32	
Rate Constant	**NUMBER**	**22**	***Coupon Spread***
Rate Factor	**NUMBER**	**22**	***Multiplier-for-Underlying-Rate***
Cap	**NUMBER**	**22**	***Rate Cap***
Floor	**NUMBER**	**22**	***Rate Floor***
Minimum Step	NUMBER	22	
Maximum Step	NUMBER	22	
Amortisation Factor	NUMBER	22	
Payout Factor	NUMBER	22	
Is Accumulating	NUMBER	22	
Underlying Instrument Identifier Type	**VARCHAR2**	**32**	***Rate-Identifier-Type***
Underlying Instrument Identifier	**VARCHAR2**	**32**	***Rate-Identifier***
Underlying Type	**VARCHAR2**	**32**	**"Underlying-Reference-Rate"**
Underlying Price Type	**VARCHAR2**	**32**	***Price Type***
Index Factor	NUMBER	22	
Index Base	NUMBER	22	
Upper Bound	NUMBER	22	
Lower Bound	NUMBER	22	
Exercise Style	VARCHAR2	32	
Mandatory Exercise	NUMBER	22	
Date Accrual Style	VARCHAR2	32	
Date Accrual Rule	**VARCHAR2**	**32**	***Accrual Rule***
Settlement Style	VARCHAR2	32	
Instrument Build	VARCHAR2	32	
Maturity Period Amount	NUMBER	22	
Protected Obligation	VARCHAR2	32	
Credit Protection	VARCHAR2	32	
Right Owner	VARCHAR2	32	
Paying Party	NUMBER	22	

Figure 5.8: Mapping a floating rate coupon to the Cash Flow Element structure.

Cash Flow Element Mapping Pattern for an Equity Call Option			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Tag	VARCHAR2	255	
Element Type	**VARCHAR2**	**32**	**"Equity Call Option"**
Element Analytic Model	VARCHAR2	32	
Element Sequence	NUMBER	22	
Element Operator	VARCHAR2	32	
Element Tag	**VARCHAR2**	**255**	***Option Reference***
Element Parent Tag	VARCHAR2	255	
Period Start Date	**DATE**	**7**	***Option Start Date***
Period End Date	**DATE**	**7**	***Last Exercise Date***
Is Detachable	NUMBER	22	
Base Instrument Identifier Type	**VARCHAR2**	**32**	**"ISO-Currency-Code"**
Base Instrument Identifier	**VARCHAR2**	**32**	***Option Currency***
Base Value	NUMBER	22	
Base Value Denominator	NUMBER	22	
Base Unit	VARCHAR2	32	
Trade Instrument	NUMBER		
Trade Instrument Identifier Type	VARCHAR2	32	
Trade Instrument Identifier	VARCHAR2	32	
Trade Value	NUMBER	22	
Trade Unit	VARCHAR2	32	
Rate Constant	**NUMBER**	**22**	***Strike Price***
Rate Factor	NUMBER	22	
Cap	NUMBER	22	
Floor	NUMBER	22	
Minimum Step	NUMBER	22	
Maximum Step	NUMBER	22	
Amortisation Factor	NUMBER	22	
Payout Factor	NUMBER	22	
Is Accumulating	NUMBER	22	
Underlying Instrument Identifier Type	**VARCHAR2**	**32**	***Underlying-Identifier-Type***
Underlying Instrument Identifier	**VARCHAR2**	**32**	***Underlying-Instrument-Identifier***
Underlying Type	**VARCHAR2**	**32**	**"Underlying-Instrument"**
Underlying Price Type	**VARCHAR2**	**32**	***Price Type***
Index Factor	NUMBER	22	
Index Base	NUMBER	22	
Upper Bound	NUMBER	22	
Lower Bound	NUMBER	22	
Exercise Style	**VARCHAR2**	***32***	**Exercise Style (e.g. European or American)**
Mandatory Exercise	**NUMBER**	***22***	**Mandatory Exercise**
Date Accrual Style	VARCHAR2	32	
Date Accrual Rule	VARCHAR2	32	
Settlement Style	**VARCHAR2**	***32***	**Settlement Style (e.g. Cash or Physical)**
Instrument Build	VARCHAR2	32	
Maturity Period Amount	NUMBER	22	
Protected Obligation	VARCHAR2	32	
Credit Protection	VARCHAR2	32	
Right Owner	VARCHAR2	32	
Paying Party	NUMBER	22	

Figure 5.9: Mapping an equity option to the Cash Flow Element structure.

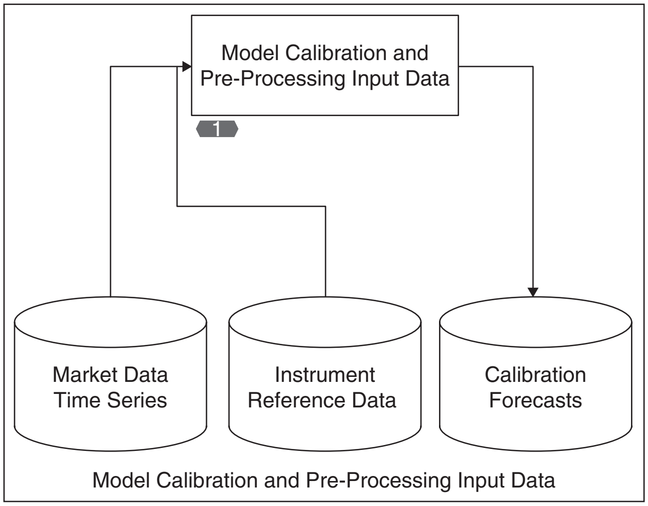


Figure 5.10: Model calibration and pre-processing input data.

Second there is the task of pre-processing input data such as a time series of market prices for an instrument and some index or time series for dividends or earnings and other data into a forecast for the price volatility of the instrument, its beta relative to the benchmark, or the growth rate of the dividend or earnings stream over time.

In some cases this pre-processing can be fully automated whereas in others it will involve considerable human input and intervention. Measuring the historical beta or volatility can be fully automated. Volatility and beta forecasts can sometimes be automated but need to at least be monitored closely. Others like earning or even dividend forecasts usually need substantial input and intervention from an analyst (team).

In your implementation you first need to determine which types of calibration and pre-processed inputs you need. Then you will need to determine how they can best be mapped into your modelling database. After that you will need to decide which type of process will create the calibration and pre-processed inputs, whether this process is part of your implementation or external to it. If the process is external to your implementation you will need to decide how you can best feed them into your database, and whether these feeds can be fully or partly automated.

If any calibration and pre-processed input data is part of your project and cannot be obtained ready to use from another source you will need to decide if their creation can be fully or partly automated, or if it should remain a manual activity for the user of your pricing models. If any can be automated you will need to ensure that they fit into your overall processes and, where necessary, be supported by tools such as input or operator screens or parameter files so that your users or model operators can adjust the processes to

Calibration and Pre-Processed Inputs

- Which Calibration and Pre-Processed Inputs are needed?
- How can they be mapped into the database?
- Which of them are part of your implementation and which not?
- How can you feed those that are external and can this be automated?
- Which of those that are not external should be fully or partly automated?

Figure 5.11: Checklist for implementing calibration and pre-processed input data.

changing circumstances. Figure 5.11 provides a checklist for deciding what calibration and pre-processing may be required which you can use as a starting point for your own projects.

Lab Exercise 5.3: Exploring Calibration and Pre-processed Input Data

1. Download the tutorial notes **LAB_5_3_ExploringCalibrationData.pdf** from the companion web site (http://modelbook.bancstreet.com/) and explore how calibration and pre-processed input data can be stored in the database, and explore common issues that arise from creating such data or using external feeds.

5.1.4 Creating Valuation Processes

Once you have your reference and market data in your database and correctly set up all instruments in terms of atomic building blocks, mapped all building blocks to your pricing models, implemented those pricing models and also compeleted any work for automating feeds or the creation of calibration and pre-processed input data, you will still need to orchestrate the whole show. The end-to-end process usually comprises many subprocesses, all of which need to be coordinated. Figure 5.12 shows a typical example of an end-to-end process for pricing financial instruments which you might use as a template for your own projects.

Although it is possible to do this coordination manually, more often than not this would be both too risky and too costly in terms of required manpower. If this is the case you will need to decide how you can implement the "Orchestration" for your valuation solution. In

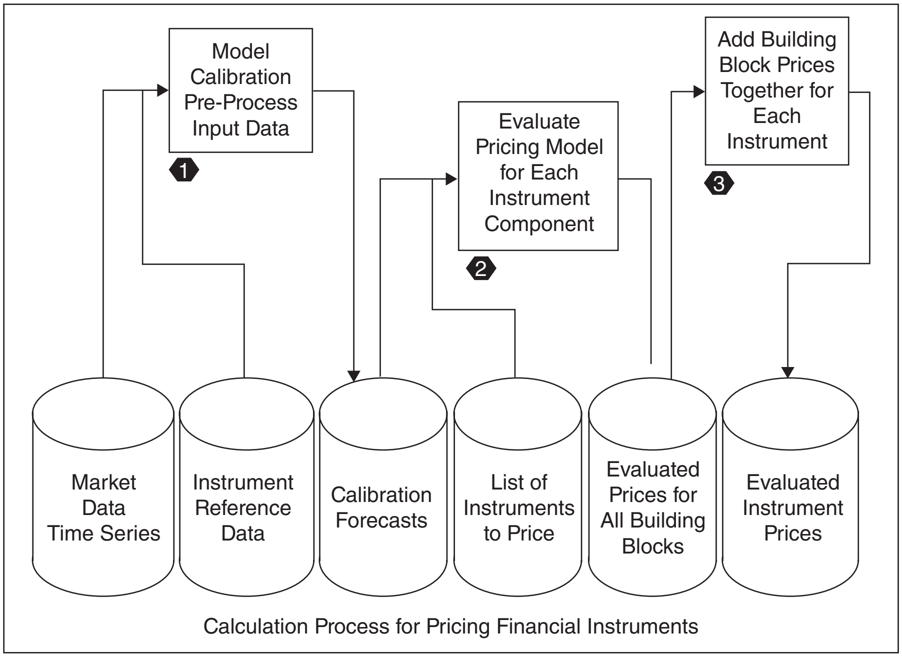


Figure 5.12: End-to-end process for pricing financial instruments.

some cases it is possible to write a simple program component or even just a script that ties all your processes together.

Usually however you will need a more flexible solution than that. One of the big benefits of the building block approach is its flexibility and the ease and speed with which you can adapt to new financial products or new approaches to pricing particular atomic building blocks. This benefit would be lost if you implement your solution as one monolithic block. In that case the whole solution would have to be reworked, retested, and reinstalled every time you want to change your high-level end-to-end process that runs your valuation models or any time you want to add or modify a single building block valuation model.

The way to avoid this predicament is to use a meta model for defining your high-level end-to-end process and then either create some software yourself that lets you execute the process as defined by the meta model or to use a readymade solution that both defines and runs your process. Figure 5.13 shows the structure of a Meta model that is used to represent the process models for analytics processes, such as creating valuations.

Before you jump into building such a solution yourself make sure you are both an expert in executable process models, the implementation of process schedulers and are a glutton for extreme punishment. The smarter way to implement your end-to-end process is to use a commercial ready-to-use solution for this. The easiest, quickest, and often the most robust solution specifically designed for the implementation of end-to-end pricing, risk, and

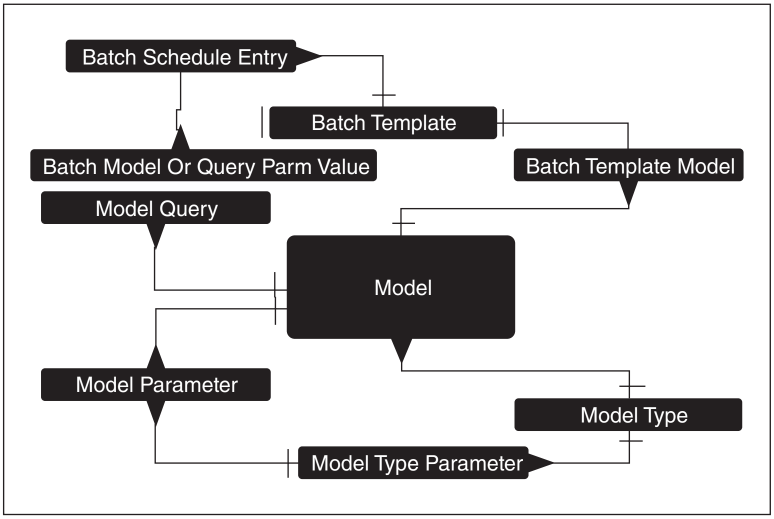


Figure 5.13: A meta model for your high-level valuations process.

performance measurement are processes such as Fincore Financial's Analytics Hub (www.fincorefinacial.com).

Alternatively you could use a generic solution such as the ORACLE business process orchestration toolkit, or use a process scheduler your organisation already uses together with any middleware that allows linking up your models, the scheduler, and your database. Although it is not as bad as building your own from scratch, this alternative still involves a substantial amount of effort, risk, and costs, and you should think very carefully about whether using a generic process orchestration toolkit or an existing general purpose scheduler together with some glue logic will be meeting your needs better.

If you use a solution specifically designed for the implementation of end-to-end pricing, risk, and performance measurement processes such as Fincore Financial's Analytics Hub, you will be able to focus on designing your models and processes rather than wasting precious time in creating the infrastructure needed to operate.

Figures 5.14 and 5.15 illustrate how easy it can be to set up your end-to-end process. Figure 5.14 shows how to register an existing or new pricing model and set up its parameters. Such parameters can be either values or the results of queries that you can enter and save in a different screen. Queries select the data according to your given criteria when the end-to-end process is run and give full control over the process.

A single valuation model could be set up many times, each time with different parameters. Each model configuration would operate on different instruments and with different calibrations or different versions of input data.

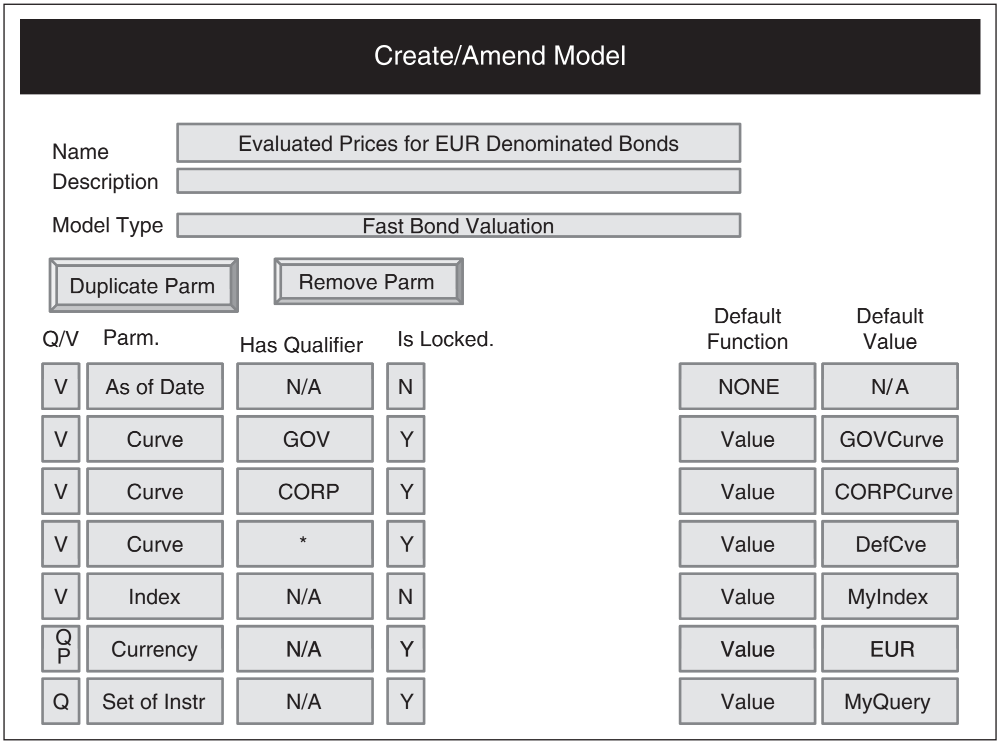


Figure 5.14: Registering valuation model components as part of a process meta model.

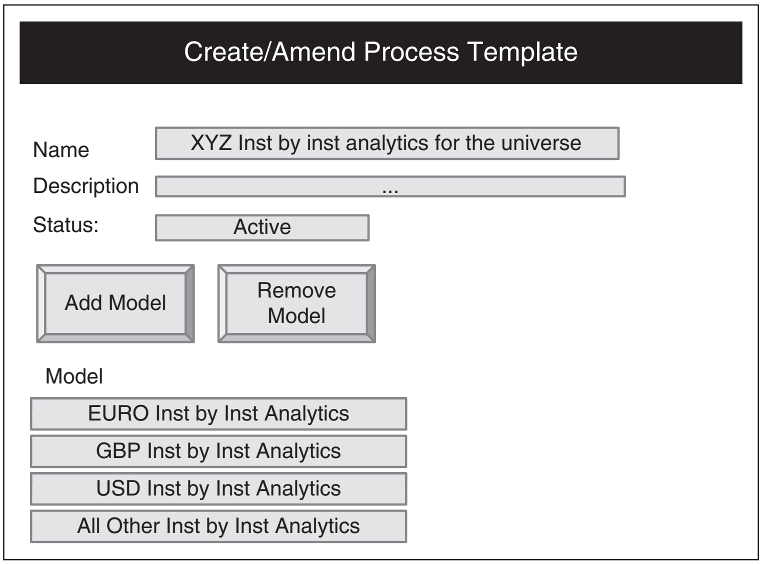


Figure 5.15: Creating a process template ready to be run using a process scheduler.

Figure 5.15 shows how you might put together different models into one overall process simply by choosing from your pool of model configurations.

Lab Exercise 5.4: Exploring End-to-End Process Configurations

1. Download the tutorial notes LAB_**5_4_EndToEndValuationProcesses.pdf** from the companion web site (http://modelbook.bancstreet.com/) to explore how you can define end-to-end valuation processes in a purpose built framework for defining and running valuation processes.

5.1.5 Refining Your Valuation Model Implementation

Model implementation, including the one of pricing models, involves a large number of assumptions and judgement calls that have to be made well before a result becomes visible. It is therefore inevitable that at least some assumptions and judgement calls will have to be revised. The fewer assumptions and judgement calls are involved and the quicker you can get to the point where you can check them against results, the easier it will be to identify what needs to be changed and to put revisions into practice.

In Chapter 1 we introduced the fountain meta model for model implementations reproduced in Figure 5.16. This provides a robust framework that helps to make sure that the necessary frequent small iterations do not veer off course or become an exercise in "extreme model hacking."

The framework also helps you demonstrate to your project stakeholders the safeguards ensuring that what you deliver is really in line with business needs. After all it is much easier to check small incremental enhancements using actual production data and processes than it is to verify big upfront promises and a large monster release tested with sample data six, 12, or 18 months down the line. Figure 5.17 provides you with a checklist for shaping the iterations which you can use as a starting point in your own projects.

In practical terms this means you should start your implementation with one or maybe two models that are easy to implement and with which you have the most experience. If the models can be mapped to many atomic building blocks, start by covering at most a handful of atomic building blocks from those that can be mapped to the models you are actually implementing. Even for pure discounted cash flow valuations this will still leave you with a challenging amount of effort.

Once you have thus determined your scope, carry your implementation through all the way to running an end-to-end process. If underway you find that you need to make further simplifications, do not be afraid to do so. Unless absolutely necessary, cut out as much

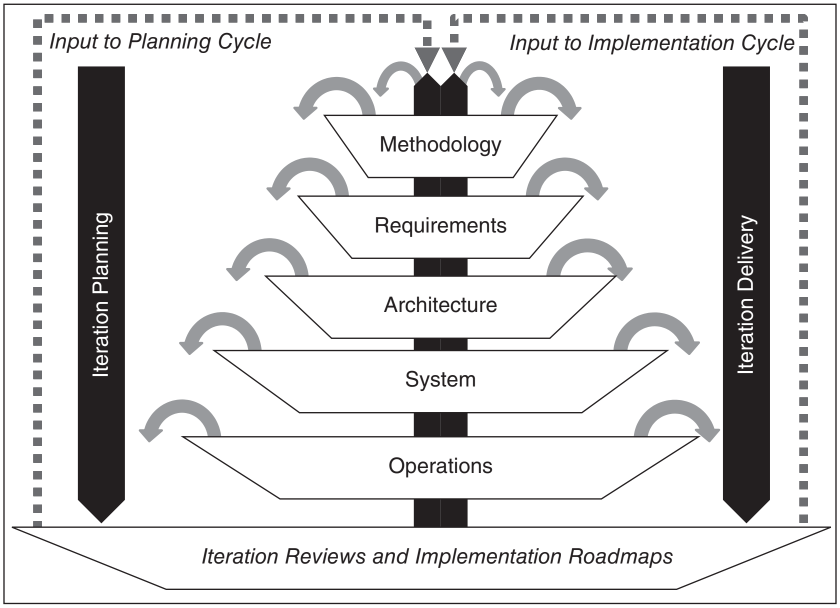


Figure 5.16: Iterative refinement of model implementations.

Small Iterations as the Road to Success

- Cut your scope to 1–2 models per iteration.
- Limit the number of atomic building blocks to 5 or less.
- Start with the easiest most familiar pricing models first.
- Use external calibration and pre-processed input data first.
- In later iterations limit the number of models for calibrations or pre-processed input data to 1–2 per iteration.
- Aim for iterations no longer than one month.
- Early iterations should be as short as possible, try one week iterations if at all possible.
- Focus on modelling rather than infrastructure – if possible use a purpose designed model implementation platform such as Fincore Financial's Analytics Hub or other similar solution.

Figure 5.17: Small iterations as the road to success.

work that could come from calibrations and pre-processing input data. Wherever possible use well-known robust external calibrations and pre-processed input data. Not only will it lighten the load and thus give you more time to get your models right and cut down the time elapsing until you can actually test your models. It will also eliminate a rich source of confusion, which can easily arise if your models for calibrations and pre-processed input data are not absolutely perfect from the start.

The first iteration or sometimes the first iterations will not result in a solution that can be put into day-to-day production even as a system running in parallel to one that you want to replace. However, you should aim to get to the point where you do have a solution that can be moved into day-to-day production in as few iterations as possible. You will gain much from the experience of having your solution in production as this experience will provide you with very valuable feedback and guidance to help shape future iterations. Many problems never become visible until your solution goes live, and the earlier in the life of a solution that you discover a problem the easier and cheaper it will be to fix.

Finally, whenever you can, try to avoid getting bogged down in infrastructure unless you want to give up model development and focus on infrastructure instead. You and your team will never have enough time to do full justice to both and the earlier you make that decision the better.

5.2 Valuation Model Implementation Checklists

In Chapter 1 we introduced a framework for model implementation. This is summarised by the charts in Figures 5.18 and 5.19. Both figures are meant to act as checklists rather than a rigid program of mandatory tasks and deliverables that have to be complete in each iteration.

You may find that, for instance, the Governance and Oversight Perspective as well as the Business Ownership perspective may receive significant attention in the first one or two iterations so that a suitable foundation for future iterations is put together. Following iterations will be much lighter in terms of effort focussing on tasks in these two perspectives since the fruits of this early effort can be used again and again in later iterations.

Then, at some point, maybe when you are about to shift gears from early prototypes and pilots to having your solution in full-scale production, the Governance and Oversight Perspective as well as the Business Ownership perspective may have to come into the foreground again as the nature of your work is shifting, and you will need to revise and extend the ground rules you had agreed earlier with your stakeholders and business sponsors.

Model Implementation Framework							
Perspective	Goals Dimension	Organisation Dimension	Schedule Dimension	Model Dimension	Data Dimension	Process Dimension	Role
Methodology Perspective	Goal setting guidelines	Organisation guidelines	Schedule guidelines	Modelling guidelines	Data guidelines	Process guidelines	Business governance
Business Perspective	Business case/plan	Organisation requirements	Schedule requirements	Model requirements	Data requirements	Process requirements	Business owner
Architecture Perspective	Architecture plan	Organisation specifications	Design schedule	Model specifications	Data specifications	Process specifications	Model designer
Systems Perspective	Build plan	Responsibility model	Build schedule	Model designs	Msg & data store design	Process designs	Model builder
Operating Perspective	Operating plan & targets	Responsibility assignments	Maintenance schedule	Production models	Production data stores	Operating schedules	Model operator
	Why?	Who?	When?	How?	What?	Where?	

Figure 5.18: Model implementation framework.

Model Implementation Methodology Template							
Set of Deliverables	Goals	Organisation	Schedule	Model	Data	Process	Persons Responsible
Methodology	Develop and implement goal setting guidelines	Develop and implement organisation guidelines	Develop and implement scheduling guidelines	Develop and implement modelling guidelines	Develop and implement data guidelines	Develop and implement process guidelines	Governance board
Requirements	Develop and implement business case & plan	Develop and implement organisation requirements	Develop and implement schedule requirements	Develop and implement model requirements	Develop and implement data requirements	Develop and implement process requirements	Business owner
Architecture	Develop and implement architecture plan	Develop and implement organisation specifications	Develop and implement design schedule	Develop and implement model specifications	Develop and implement data specifications	Develop and implement process specifications	Model designer
System	Develop and implement build plan	Develop and implement responsibility model	Develop and implement build schedule	Develop and implement model designs	Develop and implement msg & data store design	Develop and implement process designs	Model builder
Operations	Develop and implement operating plan & targets	Develop and implement responsibility assignments	Develop and implement maintenance schedule	Develop and implement production models	Develop and implement production data stores	Develop and implement operating schedules	Model operator
	Why?	Who?	When?	How?	What?	Where?	

Figure 5.19: Model implementation methodology template.

When you start planning for an iteration, Figures 5.18 and 5.19 represent ideal tools for deciding on which aspects you need to focus. Just like a pilot and co-pilot in charge of a plane go through their pre-flight checklist prior to take-off, you and your team can quickly go through the cells in each diagram and make a call of whether the item in the cell needs attention in the forthcoming iteration. Then your planning and work can concentrate on those items that matter for the current iteration.

5.2.1 Methodology Perspective—Valuation Governance Oversight

In this perspective you take a town planner's view of your valuation implementation. What is important is that you make the ground rules fit the size and nature of your endeavour. If you simply tinker with valuation models on the side, perhaps to explore their potential or better understand them, then you will need very little groundwork in this perspective. If on the other hand your valuation models are used for key financial decisions or will be used as part of a service or product, then the work that has to be undertaken in this perspective will be substantial.

The key points your planning should cover are:

- Who will use our valuations and for what purpose?
- What will happen if our valuations are wrong or unavailable?
- Who would have to answer the questions if things go wrong after this or the next few iterations? You or your team, your boss, the CEO or Chairman of the Board of your organisation?
- Have we included the right people in the dialogues or workshops for setting out the ground rules?
- Have we done enough work to make the ground rules robust enough for the current stage?
- Have we covered all the different dimensions, from Goals to Process?
- Are the ground rules just right: easy to understand, easy to put into action, and easy to measure; but complete, consistent, and comprehensive enough to rule out crack's into which you or your team, your division, or even your whole firm might fall?

5.2.2 Business Perspective—Valuation Business Ownership

In this perspective, as the name says, you need to take the perspective of a business owner. Two questions are always, or at least should always be, on a business owner's mind:

- How will this make us more profitable?
- How will this protect the business from failing?

Thus you will need to probe for how your valuations (and the way you create them) add value to your business or that of your sponsors or clients, and yet again check your valuations since when they go wrong, they could endanger your business or that of your sponsors or clients.

The key points your planning should cover are:

- Are the valuations and process for creating them fit for the intended purpose?
- Does our proposed way of creating valuations fully exploit the potential for adding value to our business or that of our sponsors or clients?
- Can we do this more cheaply, faster, with less resources, fewer errors, or in a more informative way?
- What's the biggest hole we can fall into with the chosen approach to valuation and how would we get out of it? Would we survive the fall?
- Have we got the resources to see our plan through to completion? Can we see it through beyond completion and all the way until it will be a success?

5.2.3 Architecture Perspective—Valuation Model Designer

The Designer's Perspective is dealing the decisions that set the framework for how your models will be implemented in detail. Some of these decisions may be given quantities such as may be the case if you already have a technical infrastructure for your implementation that you must use. The challenge for the designer is to shape the remaining points in the framework in such a way that both the model builder's work and the model operator's efforts together will yield the desired results both now and in the distant future.

The key points your planning should cover are:

- What is the right infrastructure for our implementation?
- Do we have the right infrastructure, and if not how can we get it?
- Will our proposed implementation work with our given data feeds and the skills and experience of the people that will operate the models?
- What is the roadmap that will deliver as much value as possible, as early as possible?
- Are we using this roadmap? If not, what stops us from using it and how can we remove such blocks?

5.2.4 Systems Perspective—Valuation Model Builder

The Builder's Perspective deals with the shop floor decision of how to actually code and put together the valuations solution. Here we can rely on a lot of ground work already laid by the other perspectives. The bigger picture in terms of what we aim to achieve, the pitfalls to avoid, and the constraints given or agreed should already be clear. Thus we can now focus on the details of the implementation.

The key points your planning should cover are:

- Are we implementing the model in the best way possible?
- Are there faster or more predictable ways at arriving at the same result?
- Are we handling all the pitfalls that our valuation model may face?
- Can and do we double-check the results? If not, is this OK under the circumstances? If the answer is yes, would my boss or the CEO or your clients or regulators agree? If they would agree now would they still agree if our valuations went horribly wrong and undetected for some time?
- If we detect a problem later or a better method becomes available, how difficult would it be to switch? If it is difficult, what can we do now to make it easier?

5.2.5 Operating Perspective—Valuation Model Operator

The Operator's Perspective deals with what it is like to use the valuation solution to produce the actual valuations for which the solution was built, under the actual operating conditions that prevailed at the time rather than what may have been assumed or planned.

The key points your planning should cover are:

- Do we fully understand the capabilities and limitations of the solution? If not where do we lack information and how could we bridge this gap?
- Can we run the solution in such a way as to produce valuations to an agreed standard with the given data feeds and the given expertise of the team?
- If not, what are the problems and what can be done in the short term to overcome them?
- How can we help the implementation team to overcome these challenges in the medium to long term?
- Are we making the best use of the solution? If not, what would we need to do to get there?

- What could go wrong with the way we operate the solution? What is the impact on the business and our clients if in the worst case? Are there ways of mitigating this? Are there ways in which the solution could be operated differently that would result in a more favourable worst case or at least reduce the impact?

5.3 More and More Advanced Pricing Models

In Chapter 5 we have introduced a basic set of commonly used pricing models covering debt instruments, equity, and simple options. There are, of course, many more in which you may be interested. Although there is no hope of covering all even in a book just on pricing models, the more models to which you have access the more you will be able to practice. We therefore have put together a lab exercise that will introduce you to more valuation models that we could not fit into this book.

Lab Exercise 5.5: Yet More Valuation Models

1. Download the tutorial notes **LAB_5_4_YetMoreValuationModels.pdf** from the companion web site (http://modelbook.bancstreet.com/) to explore models that deal with risky bonds or debt and more advanced option pricing models to price, for instance, early redemption put and call options for debt.

CHAPTER 6

Introduction to Practical Risk Modelling

In this chapter, we describe the role of models in the risk management process. We discuss the different questions for which Risk Management is asked to provide answers. The element they all have in common is the need for a measure, or set of measures, that allows the risks from positions in different asset classes or participations in different types of business to be compared to each other and to the expected and actual profits of each. In addition, the measure will need to allow decision makers to aggregate the risks at different levels of their business including an accurate summary at the very top level that includes all positions and business activities. **Value at Risk** and **Expected Tail Loss** are such measures. We examine each of them in detail. Lastly, we explain the key steps in implementing risk modelling in practice.

6.1 The Purpose of Risk Management

In general terms, **Risk Management** is concerned with understanding risks and using this knowledge to help shape a course of action that is best suited to achieve a given objective. Understanding risks means getting to understand how the outcome of a particular action can vary. Shaping a course of action involves not only deciding on an overall path but also on how to make the right decisions for each smaller step along the path toward a given goal.

In a financial or investment context, the purpose of Risk Management is to understand the range of outcomes in monetary terms from taking on particular assets or liabilities in one's portfolio. This knowledge is then used to help select combinations of assets and liabilities that best meet a particular set of objectives. As such, financial Risk Management is as applicable to running a bank as it is to running an insurance or pension fund, to the trading of financial products, or to providing investment and wealth management services to external clients. Although in each case Risk Management is not the only factor in decisions (other considerations from marketing to regulation also have to be taken into account), it plays a crucial role in each organisation's survival and success.

Financial or investment decisions often are delegated from one person or entity to another. This leads to two slightly different functions and perspectives for Risk Management when applied to investment decisions. The first function is to use Risk Management as a *tool for* *decision making*. The second is to use Risk Management as a *framework for accountability* when an investment decision is delegated by a principal to another person or entity acting as an agent on behalf of the principal. We will look at each of the two in turn.

6.1.1 Decision Making

Investment decisions are concerned with selecting particular assets or liabilities and deciding on an appropriate combination between particular assets or liabilities overall (also called portfolio optimisation; see Chapters 8 and 9).

In most cases, each asset or liability has its own risk profile and thus will provide a range of different results depending on future and as yet unknown events. It is useful to distinguish between Trading, Tactical, and Strategic Decisions since the challenges for Risk Management will be somewhat different in each case. We will briefly look at each in turn.

6.1.1.1 Trading Decisions

Trading decisions require fast decision making and are subject to constraints set by overarching tactical and strategic decisions. When deciding on a transaction, the trader faces three questions:

- How likely is it that this transaction will help maximise profits?
- How do the potential rewards of this transaction compare with the prospect of potential losses?
- How likely is it that the losses or required financial resources will be greater than some acceptable limit at any point during the life of the transaction?

Figure 6.1 illustrates the gradual refinement of decisions in the three steps set out above. Since a trading decision maker normally has the maximisation of (expected) profits as a key objective, the first question is of paramount importance to him or her. However, in order to get a reliable answer to the first question, the trader needs to have robust answers to the other two questions as well.

In particular the last question is important since although a transaction may promise outstanding profits taken on its own, it may lead to ultimate failure, either because it has the potential to wipe out the decision maker's basis for doing business or because it requires resources in excess of those the trading decision maker has or can obtain. The second question is about understanding the nature of the transaction and its likely payoffs (or losses) under different future circumstances.

Having a good answer to both questions paves the way to the answer for the first question: How likely is it that a transaction will help maximise profits? If the trader has only one

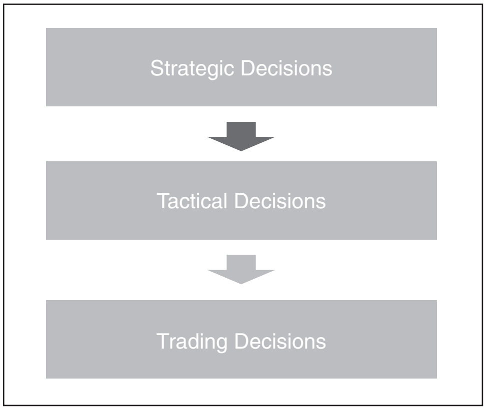


Figure 6.1: Gradual refinement of decisions.

transaction to consider, the answer to the first question is simply the combination of the answers to the second and third. If the decision maker, however, has a portfolio of assets and/or liabilities, he or she will need to use the information from the answers to the second and third questions to see how the transaction would affect his or her portfolio.

Later in this chapter we will introduce two widely used tools, **Value at Risk** (VAR) and **Expected Tail Loss** (ETL), and show you how these can be used to help answer all three questions above.

6.1.1.2 Tactical Decisions

Tactical decisions set the frame for trading decisions within given strategic constraints. The framework set by tactical decisions is usually in the form of asset or transaction type preselection (e.g., trades only in Fixed Income, Equity, or FX), budgets, and risk limits. The aim of tactical decisions is to simplify and coordinate trading decisions and to align these with the general enterprise strategy. The tactical decision maker faces three questions very similar to the ones of the trading decision maker:

- Which types of transactions will most likely help maximise profits?
- How do the potential rewards of different types of transactions compare with the prospect of any potential losses from them?
- For any relevant type of transaction, how likely is it that the losses or required financial resources will be greater than some acceptable limit at any point during the life of any such transaction?

Instead of evaluating and comparing individual transactions, tactical decision makers must evaluate different types of transactions available in their scope of business and decide whether any of them are suitable under given circumstances and what general mix of different types is most likely to lead to success. Again, the two tools, VAR and ETL, introduced later in this chapter, will be of help to provide the answer to the previous three questions.

6.1.1.3 Strategic Decisions

Strategic decisions set the frame for both tactical and trading decision making within any given organisational constraints. The framework set by strategic decision makers is usually in the form of a business model, a strategy roadmap, as well as strategic budgets and risk limits. The **business model** sets out the direction for the business, selecting certain types of transactions and ruling out others. The **strategy roadmap** sets out the expected evolution of the business model over a medium- to long-term horizon, usually three to five years.

Strategic budgets and risk limits provide concrete details for the implementation of the business model through tactical and trading decisions. They set out planned resources for achieving the objectives in the roadmap and define the organisation's risk appetite through strategic risk limits.

The strategic decision maker has to find answers to three questions, which are ultimately related to those of the tactical and trading decision makers:

- Which business model (mix of different types of transactions) will be most likely to maximise profits?
- How do the potential rewards of different business models available compare with the prospect of any potential losses from them?
- For any relevant business model(s), how likely is it that the losses or required financial resources will be greater than some acceptable limit at any point during the life of any such business model(s)?

Yet again, the two tools, VAR and ETL, will be shown to help provide the answer to these three strategic questions.

6.1.2 Accountability

When investment decisions are delegated by a **principal** to another person or entity acting as an **agent** on behalf of the principal, it is important to have a reliable and fair framework for accountability in place. Such a framework will allow the agent to demonstrate how they discharged their fiduciary duty and will allow the principal to hold the agent to account for his or her actions.

It is worthwhile distinguishing three different contexts, each requiring a different perspective on accountability. The first context is delegation in the sense of an investment mandate or of running a business unit. The second context is delegation within the context of a financial services enterprise such as a bank or insurance. Finally, the third context is delegation within the context of a society or within financial markets themselves.

We will briefly look at each context in turn.

6.1.2.1 Business Unit/Mandate

Consider a principal who delegates the management of a portfolio or a financial service business unit to another person or entity. A real life example for this is a client yielding the management of his assets to a private bank. The client most likely has clear expectations about the amount and types of risks with which he would be comfortable, and—similarly—which he would see as unsuitable or excessive risks.

The person or entity who takes on the task is equally likely to want to have clear, objective, and easy-to-interpret instructions from the principal on which types and levels of risk are acceptable and which are not.

When it comes time for the agent to render and account to the principal on how they have managed the portfolio mandate or business unit, both principal and agent are likely to want a framework that gives a clear, fair, and accurate picture of not only the results for the period in question but also the risks taken in order to obtain that result.

Later in this chapter you will see that the two mentioned tools, VAR and ETL, provide just such a framework.

6.1.2.2 Enterprise

Within a financial services enterprise such as a bank or insurance, many business units have to be coordinated and money needs to be allocated between them in such a way as to maximise the likelihood of achieving the highest possible overall return for the enterprise. At the same time, the level of risk for the enterprise as a whole should not be excessive according to pre-agreed limits.

In order to achieve this, a measure is needed that allows decision makers in different parts of the enterprise to compare the risks between proposed business from different units and add up the risks across all business units to arrive at a risk measure for the enterprise as a whole.

As we will show later, ETL is a measure that allows such risk aggregation.

6.1.2.3 Society/Market

Finally, society or the market in general may find it advantageous to put in place some means that allow controlling the risks undertaken by financial institutions. This combines with a view to minimise the risks of catastrophic collapses of financial institutions with all their negative effects they have on markets and society as a whole.

Regulators acting on behalf of governments (representing society) or of self-regulatory organs (acting on behalf of market participants) are particularly interested in ensuring that financial institutions put aside sufficient capital. This "regulatory capital" enables them to survive any unexpected but predictable extreme events, which they are likely to face throughout their lives.

In addition, regulators may find it useful for fund managers and other financial product providers to label more clearly the risks of the products they provide.

In both cases VAR and in particular ETL are useful measures that can be used to achieve the desired effect.

6.2 Early Approaches to Risk Management

Since the middle ages, when early precursors of our modern financial system emerged in the rich port and trading cities of medieval Europe (such as Florence, Genoa, Lyon, or Antwerp), the early traders and bankers were acutely aware of the importance of mastering the market and credit risks they faced for their survival and success. This even more so because at that time no social security networks existed and economic survival was closely connected to physical survival.

For much of modern history, however, Risk Management remained a "black art" that adopted new scientific advances only slowly and cautiously. One of the earliest steps toward practical scientific risk measures for investment products was the concept of duration developed by Macaulay in 1938 (see Section 6.2.1). Markowitz's seminal work on portfolio theory in the 1950s was the next big step, followed by Black, Scholes, and Merton's foundation of option pricing theory in the early 1970s.

Since then, work on approaches to market and credit risk has flourished and the "black art" of old gradually has been replaced by modern, more scientific measures. The early approaches to risk modelling prevalent in the 1970s, 1980s, and even the beginning of the 1990s can be divided into two groups:

(1) **Sensitivity Analysis**, which was more common in investment markets; and

(2) simple **Risk Simulations**, which were more common in the banking industry.

We will look at each of the two in turn.

6.2.1 Sensitivity Analysis

The most popular and influential measures of Sensitivity Analysis are best grouped by the three asset classes for which they were first developed: sensitivities for bond and money

Table 6.1: Schedule of cash flows for a hypothetical coupon bond

Term (in years)	Cash Flow in %	Present Value in %
1	10.000	9.000
2	10.000	8.110
3	10.000	7.300
4	10.000	6.570
5	10.000	5.920
6	10.000	5.330
7	110.000	52.770

market securities and portfolios, sensitivities for equity portfolios, and sensitivities for options.

6.2.1.1 Bond–Debt–Interest Rate Sensitivity Measures

The most common, useful, and popular sensitivities for interest rate instruments like bonds and money market securities as well as portfolios built from them are Duration, Modified Duration, and Present Value of an 01.

Duration is a measure of the **effective** or **discounted mean term** of a bond or other fixed income (or money market) security. It measures the expected time until the investor gets 100% of his or her money back.

The measure itself is the weighted average of cash flow terms (period measures from the date of calculation until cash flow payment) using the present value of the instrument's cash flows as weights for each of the payments up to and including redemption. Table 6.1 shows the actual and discounted cash flows of a hypothetical 7 year bond that we will use as an example to illustrate the calculation of the Duration measure.

$$\text{Duration} = \frac{\Sigma|x: 1..n|\left[\text{PresentValue}(\text{CashFlow}_x) * \text{TimeElapsed}(\text{Present}, \text{CashFlow}_x)\right]}{\Sigma|x: 1..n|\text{PresentValue}(\text{CashFlow}_x)}$$

To illustrate this concept, consider a seven-year bond with an annual 10% coupon having a price of 95.00 % and a yield of 11.063 %. The duration measure is calculated at the date of issuance of the bond. At that date, the term to the first (coupon) cash flow is 1 year, to the second 2 years, and so on. Terms are measured in year fractions.

$$\text{Duration} = \frac{\Sigma|x:1..n|\left[\text{PresentValue}(\text{CashFlow}_x) * \text{TimeElapsed}(\text{Present}, \text{CashFlow}_x)\right]}{\Sigma|x:1..n|\text{PresentValue}(\text{CashFlow}_x)}$$

$$= \frac{\Sigma 9.00*1 + 8.11*2 + 7.30*3 + 6.57*4 + 5.92*5 + 5.33*6 + 52.77*7}{\Sigma 9.00 + 8.11 + 7.30 + 6.57 + 5.92 + 5.33 + 52.77}$$

$$= 5.31 \text{ years}$$

Modified Duration is a measure of the sensitivity of the price of a bond to changes in its (Gross Redemption) Yield.

$$\text{Modified Duration} = \frac{\text{Duration}}{1 + i/n} = \frac{\text{Duration}}{\text{GrossRedemptionYield}}$$

The **Gross Redemption Yield** is the Internal Rate of Return of all cash flows to maturity, including the redemption of principal.

The Modified Duration allows you to obtain an approximation of a change in bond price given a change in yield.

Approximation of a Change in Price for a given change in yield
$$= -\text{DirtyPrice} * \text{ChangeInYield} * \text{ModifiedDuration}$$

To illustrate this concept, consider the same seven year bond as earlier, and assume a 1% change in yield from 11.063% to 12.063%:

$$\text{Modified Duration} = \frac{\text{Duration}}{1 + i/n} = \frac{\text{Duration}}{\text{GrossRedemptionYield}}$$

$$= \frac{5.31 \text{ years}}{1 + 0.11063}$$

$$= 4.78 \text{ years}$$

Using

Approximation of a Change in Price for a given change in yield
= −DirtyPrice ∗ ChangeInYield ∗ ModifiedDuration

You can calculate the approximate change in price to be

= −95.00 ∗ 0.01 ∗ 4.78
= 4.54% or **454 Basis Points**

and the new price of the bond will now be

= 90.60 (down 4.54 or 454 Basis Points from the original price of 95.00)

The **Present Value of an 01** (PV01) or **Dollar Value of an 01** (DV01) is the change in price of a bond due to a *one basis point change in yield.*

This implies the same formula as for Modified Duration, except that the change in yield is fixed to one basis point. Since a basis point equals 0.01%, the change is equal to 0.0001. Therefore,

PV01 or DV01 = ModifiedDuration ∗ DirtyPrice ∗ 0.0001

To illustrate this concept consider the same seven-year bond as before. The formula for PV01 or DV01 gives

PV01 or DV01 = ModifiedDuration ∗ DirtyPrice ∗ 0.0001
= 4.78 ∗ 95.00 ∗ 0.0001
= 0.04541

Again, you are invited to try out the above measures on the companion web site of the book using Lab Exercise 6.1 below.

Lab Exercise 6.1: Fixed Income Sensitivity

1. Open the file **FISensitivity.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the performance calculation shown above.
2. Using the R application load the file **FISensitivity.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseFISensitivity()

…

R> RunFISensitivity()

…

R> FISensitivityReport()
```

6.2.1.2 Equity Sensitivity Measures

The two most common and popular sensitivities for equity instruments and equity portfolios are the Sharpe Ratio and the Treynor Ratio.

The **Sharpe Ratio** is a measure of the Risk Adjusted Annualised Excess Return of an equity or equity portfolio. Both concepts are nested, as a one-equity portfolio is simply a single equity. We will therefore speak in the following only of equity portfolios.

The Excess Return is the return of an instrument in excess of the risk-free market rate over a given period. The Sharpe Ratio is named after its inventor, the Noble Laureate William Sharpe. The risk adjustment in the Sharpe ratio is based on the total risk of the equity or portfolio.

$$\textbf{Sharpe Ratio} = \frac{\textbf{APR} - \textbf{RFR}}{\textbf{StdDevAPR}}$$

where:

APR = Annualised Portfolio Return
RFR = Risk Free Interest Rate
StdDevAPR = Standard Deviation of the Annualised Portfolio Return

To illustrate the Sharpe Ratio, consider a portfolio with an annualised return of 9.73 with a standard deviation of the annualised portfolio return of 22.48, and assume a risk-free interest rate of 2%.

$$\text{Sharpe Ratio} = \frac{\text{APR} - \text{RFR}}{\text{StdDevAPR}} = \frac{9.73 - 2.00}{22.48} = \frac{7.73}{22.48} = 0.0344$$

The **Treynor Ratio** is an alternative measure of risk adjusted return for an equity portfolio. It replaces the risk measure of the Sharpe Ratio (Standard Deviation of the Annualised Portfolio Return) by a measuring the systematic or market risk in the form of the Beta estimated for the portfolio via the Capital Asset Pricing Model (CAPM). It is named after Jack Treynor, who first proposed it.

$$\textbf{Treynor Ratio} = \frac{\textbf{APR} - \textbf{RFR}}{\textbf{Beta}}$$

where:

APR = Annualised Portfolio Return
RFR = Risk Free Interest Rate
Beta = Systematic risk of the stock or portfolio according to the CAPM

To illustrate the Treynor Ratio, consider a portfolio with an annualised return of 9.73 and a Beta of 0.9977, and assume a risk-free interest rate of 2%.

$$\text{Treynor Ratio} = \frac{\text{APR} - \text{RFR}}{\text{Beta}} = \frac{9.73 - 2.00}{0.9977} = \frac{7.73}{0.9977} = 7.75$$

Both the Sharpe and the Treynor Ratio are relative in the sense that their absolute values do not have a direct economic meaning. They mostly serve to compare investments in differently composed equity portfolios.

We recommend that you study the usage and effect of both ratios using Lab Exercise 6.2.

Lab Exercise 6.2: Equity Sensitivity Measures

1. Open the file **EquitySensitivity.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the performance calculation shown above.
2. Using the R application load the file **EquitySensitivity.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseEquitySensitivity()

...

R> RunEquitySensitivity ()

...

R> EquitySensitivityReport()
```

6.2.1.3 Option Sensitivity Measures

Sensitivity measures for simple options can be derived from the Black & Scholes option pricing formula. The two most commonly used sensitivities are **Delta**, which measures the sensitivity of the option price to changes in the underlying price, and **Gamma**, which measures the sensitivity of the option's Delta to changes in the underlying price.

In algebraic terms, if **V** is the value of the option and **spot** is the price of the underlying being linked via

$$V = f(\text{spot}),$$

then Delta is the first derivation of V with respect to spot, and Gamma the second derivation. Gamma hence measures the speed, or velocity, by which the change in option price due to a change in underlying price increases or decreases as the changes become large and larger.

We use the Black & Scholes option pricing formula to illustrate these measures.

Black Scholes Option Pricing Formula

$$\text{Value}_{\text{Call Option}} = \text{spot} * N(d_1) - \text{strike} * N(d_2) * e^{rt}$$

$$\text{Value}_{\text{Put Option}} = -\text{spot} * N(d_1) + \text{strike} * N(d_2) * e^{rt}$$

$$d_1 = \frac{LN\left(\left(\text{spot} * e^{rt}\right)/\text{strike}\right) + \left(\sigma^2 t\right)/2}{\sigma\sqrt{t}}$$

$$d_2 = \frac{LN\left(\left(\text{spot} * e^{rt}\right)/\text{strike}\right) - \left(\sigma^2 t\right)/2}{\sigma\sqrt{t}}$$

where:

t: time to expiry as a fraction of year (365 days)
σ: annualised volatility of the underlying
r: continuously compounded interest rate
N(d): standardised normal cumulative probability function

Delta is a measure of the sensitivity of the price of an option to changes in the underlying price. It is the first derivation of V with respect to spot.

$$\text{Delta} = \frac{\text{Change in the Value of the Option}}{\text{Change in the value of the underlying}}$$

Using the Black & Scholes formula this is

$$\text{Delta}_{\text{Call Option}} = N(d_1)$$

$$\text{Delta}_{\text{Put Option}} = N(d_1)$$

where:

N(d): standardised normal cumulative probability function

$$d_1 \quad \frac{LN((\text{spot} * e^{rt})/\text{strike}) + (\sigma^2 t)/2}{\sigma\sqrt{t}}$$

Gamma is a measure of the sensitivity of the delta of an option to increasing changes in the price of the underlying asset. It is the second derivative of V with respect to spot.

$$\textbf{Gamma} = \frac{\text{Change in the Delta of the Option}}{\text{Change in the value of the underlying}}$$

Using the Black & Scholes formula this is

$$\textbf{Gamma} = \frac{1}{S\sigma\sqrt{\pi t}e^{Z}}$$

where:

$$Z = \frac{(d_1)^2}{2}$$

N(d): standardised normal cumulative probability function

$$d_1 = \frac{LN\left((spot * e^{rt})/strike\right) + (\sigma^2 t)/2}{\sigma\sqrt{t}}$$

The usage of option sensitivity measures requires some exercise, which you can get by trying out Lab Exercise 6.3 below.

Lab Exercise 6.3: Option Sensitivity Measures

1. Open the file **OptionSensitivity.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the performance calculation shown above.
2. Using the R application load the file **OptionSensitivity.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseOptionSensitivity()

...

R> RunOptionSensitivity ()

...

R> OptionSensitivityReport()
```

6.2.2 Risk Simulation

Sensitivities are helpful when comparing individual assets or portfolios but they do not give much insight into the variation of risk through time or under specific circumstances. In order to fill this gap, practitioners have developed a variety of simulation approaches. Two approaches, **Gap Analysis** and **Stylised Scenarios**, have found particularly widespread use and we therefore present each of them in the following sections.

6.2.2.1 Gap Analysis

Gap Analysis has its root in the need by commercial banks to match their lending at different maturities with both equity and debt funding at appropriate maturities in such a way that the bank will remain solvent and profitable.

A Gap Analysis worksheet shows the period-by-period and cumulative gap between an organisation's financial assets and its liabilities. Although Gap Analysis originated in commercial banks, it is useful for any entity with material and nontrivial financial asset and liability sections in their actual or notional balance sheet.

The technique is best illustrated by way of a worked example.

Assume a newly opened bank with a balance at the end of day one as shown in Table 6.2. The bank has raised a total of $10,000 USD, split between $3,000 USD in equity and $7,000 USD through issuing certificates of deposit in its own name in the money market.

The bank has invested these funds by making a variable rate loan to a credit high quality business borrower maturing in 15 years' time. The terms of the loan allow the bank to reset the annualized interest rate for the loan every six months, at the prevailing interest rate for Certificates of Deposits plus a spread of 150 basis points, or 1.5%.

In the Gap Analysis, the CDs are recorded in the six-month asset bucket. The commercial loan is also shown in the six-month bucket since its interest rate will reset to the interest rate of six-month instruments every six months. The gap of $3,000 USD at the six-month bucket column reflects the equity funds of $3,000 USD. Equity is not recorded as a liability with a specific maturity but should always be reflected through the interest rate gap.

Although crude, Gap Analysis is very useful in illustrating, at a high level, an organisation's risk exposure to interest rates. Table 6.3 shows the detailed workings for the Gap Analysis example.

Table 6.2: Balance sheet

Item	Notes	Amount
Assets		
Deposits with Other Banks		—
Commercial Loans	1	10,000
Retail Loans		—
		10,000
Liabilities		
Deposits from Customers		—
Short Term Borrowing	2	7,000
Long Term Borrowing		—
Equity	3	3,000
		10,000

Notes
1. Long-term (15 years) variable rate business loan with an interest rate resetting every 6 months to the 10 current spot market rate for USD denominated 6 Month Certificates of Deposit (CD) plus a 150 Basis Point spread.
2. Issuance of Certificates of Deposit (CD) maturing in six months' time.
3. Equity Shares issued for cash.

6.2.2.2 Stylised Scenario Analysis

The use of Stylised Scenario Analysis started to grow with the increasing availability of ever cheaper and ever more powerful computer hardware. Stylised Scenario Analysis involves the simulation of the financial status of an organisation under different hypothetical scenarios.

Although particularly popular with banks, this technique is equally applicable to investment portfolios, insurance, or pension funds. Stylised Scenarios are a key element for stress testing where the scenarios focus on extreme events rather than more likely expected changes.

Similar to Gap Analysis, Stylised Scenario Analysis is best illustrated using a worked example.

Assume a bank with a balance sheet and income statement sheet as shown in column 01 in Table 6.4 on page 157. To see how the balance sheet and financial position of the bank would look under different circumstances, the bank needs to go through a number of steps.

Step 1: The bank will have to define a range of different scenarios. In the example below, some stylised scenarios are shown in columns 02 to 09 and are briefly characterised by descriptive titles such as "3% Yield curve shift downward." Each scenario needs to be cast into a form that can be used as parameters for the numerical models that implement the simulation.

Step 2: The bank will have to build a model for how their positions in different assets may change under different assumptions. Will lending to business or retail consumers for, say, a given fixed rate or for a certain variable rate loan product go up or down with a rate increase, and if so by how much? Each such question must be cast into a form that can be integrated into a numerical simulation model ultimately covering all assets and liabilities.

Step 3: The bank will have to build a numerical valuation and income estimation models for all assets that are to be included in the calculations.

Step 4: Finally, the bank will have to run the simulation model from step 2 using the different parameters for each scenario from step 1. The results then can be used as the input for the model from step 3 together with the parameters from step 1 for the corresponding scenarios.

Step 5: After running the model from step 3 for each scenario, the results for all scenarios from steps 1 to 3 can be summarised and analysed in suitable reports such as the one used as an illustration in this example.

■

6.3 Modern Approaches to Risk Management

Although many of the earlier approaches such as Sensitivities, Gap Analysis, or Stylised Scenario Analysis still have their usefulness, none of them can easily be applied across all types of risks to which a financial institution or other organisation is exposed.

In the first part of this chapter we looked at the different questions for which Risk Management is asked to provide answers. The element they all have in common is the need for a measure that can be used across different asset classes or different types of financial services business.

This measure, or set of measures, should allow the risks from positions in different asset classes or participation in different types of business to be compared to each other and to the expected and actual profits of each. In addition, the measure will need to allow decision makers to aggregate the risks at different levels of their business, including an accurate summary at the very top level that includes all positions and business activities.

Value at Risk (VAR) and **Expected Tail Loss** (ETL) are such measures and we will now look at each in more detail.

Table 6.3: Gap analysis

Interest Sensitivity Gap Analysis												
Item	**1 Day**	**8–30 Days**	**31–60 Days**	**61–90 Days**	**3–6 Mth**	**7–9 Mth**	**10–12 Mth**	**2–4 Yrs**	**5–7 Yrs**	**8–10 Yrs**	**11+ Yrs**	**TOTAL**
Assets												
Bank Deposits												0
Commercial Loans					10000							10000
Retail Loans												0
	0	0	0	0	10000	0	0	0	0	0	0	10000
Liabilities												
Customers Deposits												0
Short Term Borrowing					7000							7000
Long Term Borrowing												0
	0	0	0	0	7000	0	0	0	0	0	0	7000
Interest Rate Gap	0	0	0	0	**3000**	0	0	0	0	0	0	**3000**
Cumulative IR Gap	0	0	0	0	3000	3000	3000	3000	3000	3000	3000	3000

Table 6.4: Scenario simulation summary

Scenario Simulation Summary for Sample Bank Inc.										
Item	**01**	**02**	**03**	**04**	**05**	**06**	**07**	**08**	**09**	**10**
	Baseline	3% Yield curve shift downward	3% Yield curve shift upward	6% Yield curve shift upward	3% YC twist up at long term	6% YC twist up at long term	3% YC twist up at short term	6% YC twist up at short term	10 Drop in Equity Mkts	...
Assets										
MM Assets	65	12	92	127	114	78	73	87	54	...
Commercial Loans	100	120	80	60	60	90	90	110	95	...
Retail Loans	27	80	20	18	29	23	19	15	33	...
Others	8	8	8	5	7	9	8	3	8	
	200	220	200	210	210	200	190	215	190	...
Liabilities										
Customers Deposits	60	40	60	50	40	60	60	90	60	...
Short Term Borrowing	20	60	20	40	50	20	10	5	10	...
Long Term Borrowing	80	80	80	80	80	80	80	80	80	
Equity	40	40	40	40	40	40	40	40	40	...
	200	220	200	210	210	200	190	215	190	...
Income										
Interest Income	10	9	11	12	9	8	11.5	13	9	...
Fee Income	9	11	9	8	11	12	8	5	3	...
Other Income	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	...
Admin Expenses	(11)	(9)	(10)	(12)	(9)	(8)	(10)	(11)	(7)	...
Net Income	8.9	11.9	10.9	8.9	11.9	12.9	10.4	7.9	5.9	...

Note: All numbers are in terms of millions of USD

Table 6.5: Stock of an initial holding of 10 stock at 8.55 EUR on date t

Day t+i	Price	Value of Position	P/L Result in EUR as compared to value at date t
1	8,55 EUR	855 EUR	N/A
2	8.78 EUR	878 EUR	+23,00 EUR
3	8.75 EUR	875 EUR	–3,00 EUR
...	...	...	...
95	8.25 EUR	825 EUR	–11,00 EUR
96	8.30 EUR	830 EUR	+5,00 EUR
97	8.32 EUR	832 EUR	–3,00 EUR
98	8.28 EUR	828 EUR	–4,00 EUR
99	8.22 EUR	822 EUR	–6,00 EUR
100	8.29 EUR	829 EUR	+7,00 EUR

6.3.1 Value At Risk (VAR)

Value at Risk (VAR) is a measure that reports the maximum loss from holding an asset (or carrying on a business activity) over a given period and for a given probability level. The easiest way to illustrate what this means is by way of example.

Assume that we are buying 10 shares of a company listed on the Euronext Stock Exchange (Paris) at the end-of-day price of that stock exchange of 8.55 EUR.

The value of our holding at the end of the first day is therefore 10 stocks at 8.65 EUR or 865.00 EUR in total. We then record the end-of-day price of that stock for each day thereafter for exactly 100 days. Table 6.5 shows the evolution of our position over time assuming that we do not change the number of stocks we hold.

All Profit/Loss (P/L) calculations are done with reference to previous day.
In our example we have:

- On date t = 2 a stock price of 8.78 EUR per share, leading to a position value of 878.00 EUR and a hypothetical (as not realised) profit of +23.00 EUR on the portfolio compared to date t = 1.
- On date t = 3 a stock price of 8.75 EUR per share, leading to a position value of 875.00 EUR and a hypothetical (as not realised) loss of −3.00 EUR on the portfolio compared to date t = 2.
- On date t + 100 a stock price of 8.29 EUR per share, leading to a position value of 82.00 EUR and a hypothetical (as not realised) profit of + 7.00 EUR on the portfolio compared to date t = 99.

With these numbers we are able to do simple VAR calculations. Because our valuations and hence our P/L were calculated at a daily basis, our data constrains us to a VAR measure for a single day.

If we now sort the 100 P/L results in descending order, like the one shown in Table 6.6, we can simply read the one day VAR for a given probability level by going to the corresponding row, reading the P/L and multiplying it by –1.

The VAR at 95% probability is 10.00 USD. Thus according to our table at a probability of 95% the most we can lose over a day is just over $5 USD or $5.30 USD, precisely.

The beauty of VAR is that we can calculate it for any type of asset class or financial business. We can even calculate it for portfolios of assets or business units. Thus, with VAR we have a measure of risk that can span entire portfolios or enterprises of arbitrary complexity.

VAR of a Portfolio IS NOT EQUAL to the sum of the VAR of its components.

One problem that remains with VAR is that it is not possible simply to obtain the VAR of a portfolio or an enterprise once we have the VAR measures of its component parts. The VAR of a portfolio or enterprise is likely to be very different from the sum of the VAR measures of its parts. The reason is that VAR measures of various parts are likely to be

Table 6.6: One-day VAR table for a hypothetical equity position

Row No	P/L Result (USD) = –VAR for a given Probability	Probability Level (P)
1	25.5	1%
2	21.9	2%
3	18.7	3%
...	...	...
95	–10.0	95%
96	–11.8	96%
97	–13.0	97%
98	–13.3	98%
99	–13.7	99%
100	–20.0	100%

To calculate VAR at a given level, just read the P/L result from the corresponding row and multiply by –1; for example, one day VAR at 95% is –10.00 EUR *–1 or simply 10.00 EUR, and VAR at 99% is 20.00 EUR.

(positively or negatively) correlated. If part-specific VAR measures were uncorrelated, or independent of each other, then the VAR of the aggregate would be the VAR of the sum of its parts. Otherwise, the aggregate VAR measure depends on the correlation matrix between the VAR measures of its parts.

The effect of this is that in some cases, the VAR of a portfolio or enterprise can be significantly higher than the sum of the VAR of its parts, whereas in others it will be significantly less.

If you are now tempted to rush off and implement the method set out in the previous example, you should wait a little. Although the example is correct, it has been constructed to make it simple and easy to understand, and a lot of real-life practicalities are not covered. In Chapter 7 we will look at three different ways of calculating VAR in real-life practice:

6.3.1.1 Historical VAR

Historical VAR is very similar to the method we just used in our example, with the difference that Historical VAR allows us to deal with much bigger data sets than 100 days, and to calculate VAR for horizons other than one day.

6.3.1.2 Parametric VAR

The second method, Parametric VAR, is a simplification in terms of calculation effort that can be used in many circumstances. It reduces the need of keeping extensive data sets of historical P/L.

Instead of "reading off" VAR from tables of historical results, Parametric VAR works by first obtaining parametric statistics (such as mean or variance) that describe the tables of historical results. Having these at hand allows calculating the VAR measures through the use of statistical formulas without the need to refer back to or keep the original tables of results.

6.3.1.3 Simulation VAR

The third method, Simulation VAR, is similar to Historical VAR, but instead of using historical results it uses simulation methods like Monte Carlo or Quasi Monte Carlo Simulation to generate the tables of results using parametric statistics and estimations as an input.

Although at first sight this may look like a wasteful combination, it actually is a vital tool to extend VAR into areas where both Historical and Parametric VAR run into unsolvable practical problems.

> **VAR measures the largest loss only at a given probability but ignores any other and potentially much bigger and more devastating losses in the remainder of the results.**

Before we conclude this first introduction to VAR, we need to look at one further important drawback of VAR.

Since VAR gives us only the predicted maximum loss at the given probability, it ignores any bigger losses in the result set. These losses can potentially be much bigger and much more devastating than the loss suggested by the VAR. When using VAR this should always be borne in mind.

However, VAR has a close cousin, Expected Tail Loss (ETL), that overcomes this drawback. The final part of this chapter will give you a brief introduction to ETL.

Lab Exercise 6.4: Simple VAR

1. Open the file **SimpleVAR.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the performance calculation shown above.
2. Using the R application load the file **SimpleVAR.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseSimpleVAR()

…

R> RunSimpleVAR()

…

R> SimpleVARReport()
```

6.3.2 *Expected Tail Loss (ETL)*

Expected Tail Loss (ETL) is a measure that tells us our expected maximum loss from holding an asset (or carrying on a business activity) over a given period and for a given probability level. The easiest way to illustrate what this means is by resuming our simple example from the section on VAR measures.

Assume you have £100 to invest, and invest them in a portfolio consisting of several stocks and a small amount of cash. Again perform the same procedure as in the example in Section 6.3.1 and observe the mark-to-market values of the portfolio and hence the day-on-day profit or losses. Then sort them by size and you obtain a table like profits and losses sorted from biggest profit to largest loss: This is shown in Table 6.7.

In order to create a "ready reckoner," or table from which we can read ETL, we need to create a new table very similar to the one we already have but with one important difference.

In the column where we had the P/L we put the ETL for the given probability level. This is calculated from the VAR table by summing all P/L from the row number concerned down to row number 100, dividing the sum by the number of rows added up and multiplying the result by −1.

If we carry out this procedure with the data from the earlier VAR table, we can obtain the ETL table shown in Table 6.8.

Now we can again simply read off the ETL for a desired probability level. Using Table 6.8 we can easily see that the one-day ETL for our hypothetical equity position at 95% is 7.35 EUR and ETL at 99% is 10.40 EUR.

Like with VAR, this example is a simplification. To get you started with real-life modelling of ETL, the next chapter will also look at three different practical ways of calculating ETL, which are analogous to the corresponding VAR calculation measures, Historical ETL, Parametric ETL, and Simulation ETL.

Table 6.7: One-day VAR table for a hypothetical equity position

Row No	P/L Result (USD) = −VAR for a Given Probability	Probability Level (P)
1	25.5	1%
2	21.9	2%
3	18.7	3%
...	...	...
95	−10.0	95%
96	−11.8	96%
97	−13.0	97%
98	−13.3	98%
99	−13.7	99%
100	−20.0	100%

To calculate VAR at a given level, just read the P/L result from the corresponding row and multiply by −1; for example, one-day VAR at 95% is −10.00 EUR *−1 or simply 10.00 EUR, and VAR at 99% is 20.00 EUR.

Table 6.8: One-day ETL

One-day ETL Table for a Hypothetical Equity Position		
Row No	**ETL for a Given Probability**	**Probability Level (P)**
...	...	...
...	...	...
95	7.35	95%
96	7.76	96%
97	8.25	97%
98	9.03	98%
99	10.40	99%
100	11.10	100%

To obtain ETL for this position at a given probability level, read the ETL from the corresponding row; for example, one-day ETL for our hypothetical equity position at 95% is 7.35 EUR and ETL at 99% is 10.40 EUR.

6.3.2.1 Historical ETL

The first method, Historical ETL, is again very similar to the method we used in our previous example with the difference that Historical ETL allows us to deal with much bigger data sets than 100 days and to calculate ETL for horizons other than one day.

6.3.2.2 Parametric ETL

The second method, Parametric ETL, is again a simplification that can be used in many circumstances and that reduces the need of keeping extensive data sets of historical P/L.

Instead of "reading off" ETL from tables of historical results Parametric ETL works by first obtaining parametric statistics that describe the tables of historical results and then calculating the ETL through the use of a formula from the statistics without the need to refer back to or keep the original tables of results.

6.3.2.3 Simulation ETL

The third method, simulation ETL, is yet again similar to the first but instead of using historical results it uses simulation methods like Monte Carlo and Quasi Monte Carlo Simulation to generate the tables of results using parametric statistics and estimations as an input. Although at first sight this may look like a wasteful combination, it actually is a vital tool to extend ETL into areas where both Historical and Parametric ETL run into unsolvable practical problems.

Lab Exercise 6.5: Simple ETL

1. Open the file **SimpleETL.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the performance calculation shown above.
2. Using the R application load the file **SimpleETL.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseSimpleETL()

...

R> RunSimpleETL()

...

R> SimpleETLReport()
```

CHAPTER 7

Implementing Risk Models

In this chapter we demonstrate step by step how to build and implement risk models in an evolutionary iterative way within a well-organised framework. We show how to create and use checklists to shape and support the risk model implementation effort, and how to master the meta-, input, and output data that you need for your risk models within the framework of the data model blueprint introduced earlier.

7.1 Risk Model Implementation—Step by Step

When implementing risk models it is helpful to follow and implement a number of logical steps. We distinguish five main steps, depicted in Figure 7.1, that will be discussed at full length in the subsequent sections.

1. Compose the Risk Model Structure: Decide on how your exposures are to be aggregated from individual holdings through a portfolio hierarchy structure into an ultimate global exposure.
2. Map Building Blocks to Risk Models: Create the risk models themselves and then map the risk models to individual instrument building blocks. This can be as simple as adopting a standard palette of risk model components and atomic building blocks or may involve you in developing a palette of specialised building blocks and models yourself.
3. Create and Calibrate Risk Model Inputs: Either identify some suitable source or build models to create your own inputs where they need to be transformed before they can be used in your valuation models. Typical examples of such inputs are market yield curves, beta values for companies, or equity instruments and volatilities of the prices of instruments used as underlying in an option. A special, crucial input is the variance/covariance matrix for all assets under analysis if you are using either VAR or ETL.
4. Create Risk Modelling Process: Create an overall process that runs your risk models, feed them with data, and write any results back into the database so that later they can be used in online displays, reports, and data feeds.

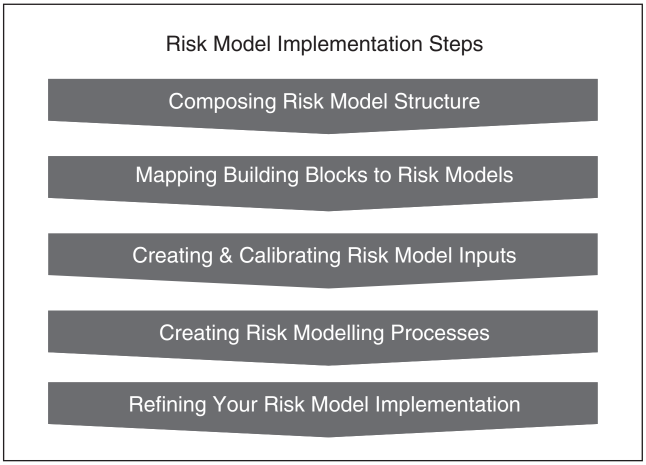


Figure 7.1: Risk model implementation steps.

5. Refine the Risk Model Implementation: Even in simple cases you should start with only a very small set of risk models, ideally just one, then go on to refine your implementation in later iterations. You can thus add additional models and input data step by step in future iterations once you have the basic process running to a required standard.

Let us now look at each step in turn and then consider how best to shape the gradual refinement of your model implementation.

7.1.1 Composing the Risk Model Structure

As a first step in your risk model implementation you will need to determine how to aggregate the exposures from different business activities in a hierarchy that ultimately combines all exposures into one total exposure. Grouping your exposures from the bottom up will allow you to reflect organisational accountability and limits for risk from traders and desks up to the level of the entire organisation.

Trader positions are rolled up into desk-level positions, which in turn are then rolled up by product group or division and so on until you finally arrive at the organisation level.

7.1.1.1 Basic Exposures at Trader or Desk Level

Figure 7.2 shows the roll up of individual exposures from trader/desk level portfolios at the bottom, each with the individual holdings up to the level of the total organisation. In the example, the trader/desk level portfolios are split by product type. Each trader/desk thus has

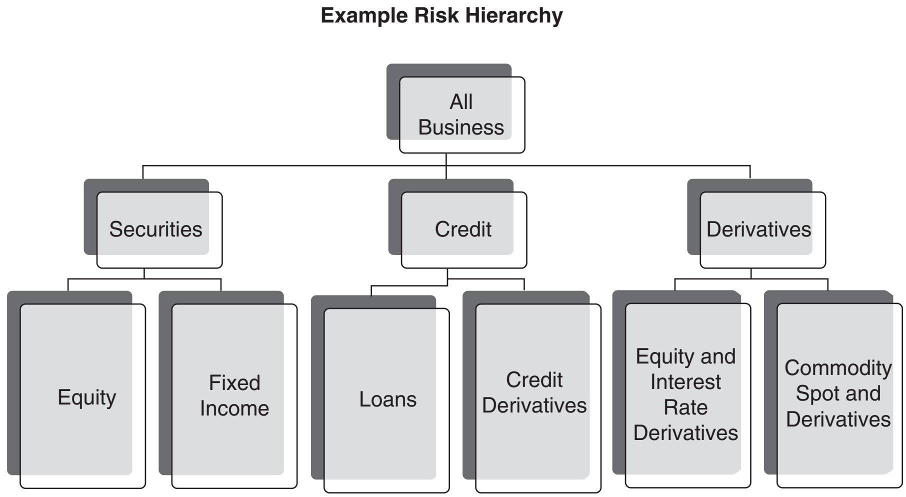


Figure 7.2: Example risk exposure aggregation hierarchy with three layers.

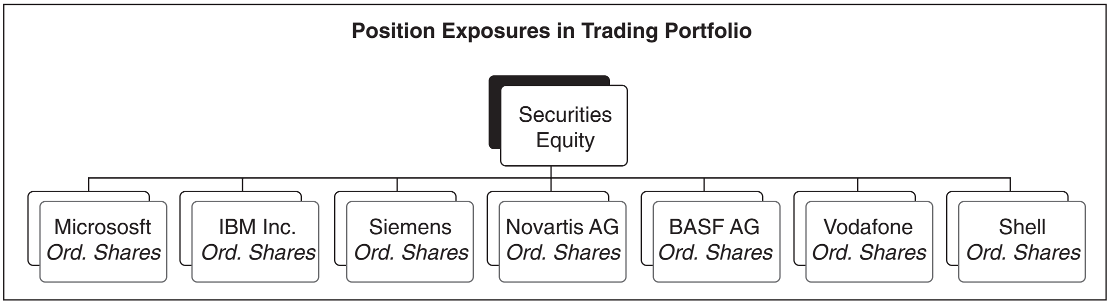


Figure 7.3: Position exposures in a trading portfolio.

a trading portfolio that contains only products of a specific type like equities, fixed income securities, loans, and the like.

For example, the trader/desk portfolio SECURITIES-Equity should only contain positions in equity instruments.

In Figure 7.3 you can see the trader/desk portfolio SECURITIES-Equity expanded to show the holdings in the portfolio in form of a tree diagram. Each leaf in the tree in Figure 7.3 will need to be represented by a position record in the database using the record structure set out in Figure 7.4.

In Figure 7.5 you can see an actual example for a trading position in the ordinary shares of Microsoft Inc. in the trader/desk portfolio SECURITIES-Equity.

Portfolio Positions			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Portfolio Instrument	NUMBER		ID of the portfolio
Portfolio Version	VARCHAR2	32	ID of the portfolio snapshot or version
Constituent Instrument	NUMBER		ID of the instrument held in the portfolio
Quantity	NUMBER	22	Long (+) or Short (-) position held
Valuation	NUMBER	22	Either the cost, valuation, or period return for the position
Effective Date	DATE	7	Date of last revision to the position
Position Transaction Reference	NUMBER		Cross reference to the transaction that established the position
Position Function	VARCHAR2	32	Type of position
Position Stage	VARCHAR2	32	Type of position
Currency	VARCHAR2	3	Currency of the position
Is Valid	NUMBER	22	Indicates whether the position is still in use for current or historical reporting or has been entirely superseded
Valid From Date	DATE	7	Date from when the position existed or the beginning of a period from when to measure the return on the position
Valid To Date	DATE	7	Last date when the position existed or the end of a period from when to measure the return on the position

Figure 7.4: Structure for position records for base level and aggregate positions.

Detailed positions from this level give the actual exposures to individual instruments. In Figure 7.6 you can see how the same record can be used to record the Mark-to-Market Profit or Loss over a given holding period for the position from Figure 7.5.

7.1.1.2 Aggregate Exposures

To analyse risk across your business you will need to aggregate exposures all the way up to the level where you have a total aggregated portfolio covering your entire business as shown in Figure 7.2.

You can build up aggregate exposures using the same machinery as that which you use to record and track exposures at the trader or desk level. Each aggregate portfolio is there again set up as a portfolio instrument itself.

Portfolio Positions			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**Example Value**
Portfolio Instrument	NUMBER		0000011
Portfolio Version	VARCHAR2	32	CURNETPOS
Constituent Instrument	NUMBER		5456789 Microsoft Ordinary Shares
Quantity	NUMBER	22	778,000
Original Cost	NUMBER	22	3,788,789
Effective Date	DATE	7	07/10/2008
Position Transaction Reference	NUMBER		9000990102
Position Function	VARCHAR2	32	Running Position
Position Stage	VARCHAR2	32	4 - Settled
Currency	VARCHAR2	3	CHF
Is Valid	NUMBER	22	True
Valid From Date	DATE	7	07/10/2008
Valid To Date	DATE	7	

Figure 7.5: An example of position record for a base level trading position.

Portfolio Positions			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**Example Value**
Portfolio Instrument	NUMBER		0000011
Portfolio Version	VARCHAR2	32	M2MPNL_20080710_V20080710
Constituent Instrument	NUMBER		5456789 Microsoft Ordinary Shares
Quantity	NUMBER	22	778,000
Original Cost	NUMBER	22	–25,678
Effective Date	DATE	7	07/10/2008
Position Transaction Reference	NUMBER		89999990102
Position Function	VARCHAR2	32	Mark to Market Profit / Loss Snapshot
Position Stage	VARCHAR2	32	4 - Settled
Currency	VARCHAR2	3	CHF
Is Valid	NUMBER	22	True
Valid From Date	DATE	7	06/10/2008
Valid To Date	DATE	7	07/10/2008

Figure 7.6: Profit/loss for a given trader/desk level trading position.

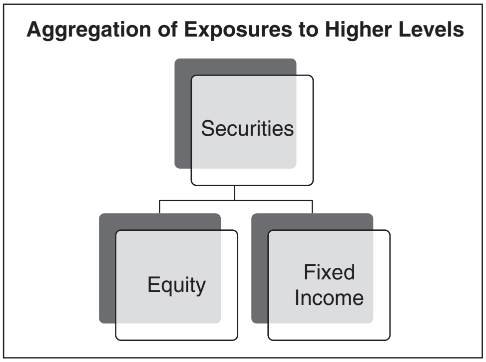


Figure 7.7: Aggregation of exposures to higher levels.

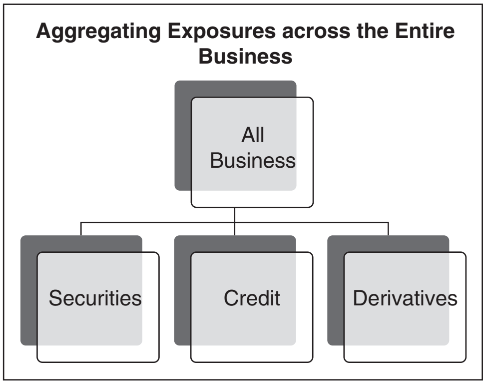


Figure 7.8: Aggregating exposures to the level covering the entire business.

The aggregate portfolio SECURITIES in Figure 7.7 can thus be defined as a portfolio instrument and the trader/desk level portfolios SECURITIES–Equity and SECURITIES–Fixed-Income can be linked to the composite via a special running position record for each.

The example running position record linking the base portfolio SECURITIES–Equity to the Composite portfolio SECURITIES can be seen in Figure 7.8.

Analogously, the Global Composite Portfolio in Figure 9.10 (of Chapter 9) can yet again be built the same way as the composite portfolio EUROPE. The All Business aggregate exposure portfolio should thus be set up as a portfolio instrument. Then each of the component aggregates, SECURITIES, CREDIT, and DERIVATIVES can be linked to it using a portfolio position record similar to the one in Figure 7.9.

Once the exposure aggregation hierarchy is completely set up it is possible to create detailed roll-up exposures for each instrument from individual trader/desk level positions to instrument-by-instrument aggregate positions at the firm level.

Portfolio Positions			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**Example Value**
Portfolio Instrument	NUMBER		0000025 SECURTIES
Portfolio Version	VARCHAR2	32	PTFHIER
Constituent Instrument	NUMBER		0000011 SECURITITES - Equity
Quantity	NUMBER	22	1
Original Cost	NUMBER	22	
Effective Date	DATE	7	15/03/2008
Position Transaction Reference	NUMBER		2300990102
Position Function	VARCHAR2	32	Portfolio Composite
Position Stage	VARCHAR2	32	4 - Settled
Currency	VARCHAR2	3	CHF
Is Valid	NUMBER	22	True
Valid From Date	DATE	7	23/09/2007
Valid To Date	DATE	7	

Figure 7.9: Position record linking lower level exposures to a higher aggregate level.

Mapping an Equity Call Option to an Option Risk Model	
Model Parameter	**Source**
t = time to expiry as a fraction of year (365 days)	*Number of years between valuation date and redemption Date*
σ = annualised volatility of the underlying	*Obtain from Instrument Analytics table*
r = continuously compounded interest rate	*Read this from a suitable yield curve*
Strike price	*This is the Strike Price term*
Current spot price	*Obtain from Instrument Price table*

Figure 7.10: Mapping an equity call option to the Black & Scholes option pricing model.

These positions from trader/desk level upward give you the instrument exposures you will need to calculate VAR or ETL for any given portfolio at any level in the hierarchy.

7.1.2 Mapping Building Blocks to Risk Models

The second main step in a risk model implementation is to map the information in your instrument database to your risk models. Mappings in risk models are similar to or extend those you have already created for instrument pricing models.

In Figure 7.10 you can see a mapping that shows how to obtain the parameters for a Parametric VAR/ETL model from the terms of an Equity Call Option building block such as the one given in Figure 7.13. Some parameters such as the **Strike Price** can be read directly from the terms of the Equity Call Option building block whereas others such as the Volatility of the Underlying or the current Spot Price can be looked up in other parts of the database with the help of the terms of the Equity Call Option building block. In Figure 7.11 and 7.12 you can see the description of a bullet redemption and floating rate coupon feature respectively which together would define the key parts of the financial characteristics of a floating rate bond as needed as input for a risk model.

A few remaining ones like the riskless interest rate and the valuation date need to be set up when the model is configured for a particular purpose or given to the model when it is actually run.

We recommend to deepen your intuition about risk model mapping using Lab Exercise 7.1.

Lab Exercise 7.1: Exploring Risk Model Mappings

1. Download the tutorial notes **LAB_7_1_ExploringRiskModelMappings.pdf** from the companion web site (http://modelbook.bancstreet.com/) to see and explore further model risk mappings like the ones shown in this chapter, but for other common building blocks like equities, floating rate coupons, and index linked redemptions.

7.1.3 Creating and Calibrating Risk Model Inputs

Many model inputs, like the coupon rate of a fixed coupon bond or the strike price of an option, are static reference data and can be obtained from the terms of instrument building blocks like coupons, redemptions and options. Nevertheless, all risk models also need further inputs related either to the instrument itself, its issuer, or the market more generally. Figure 7.14 illustrates how the model calibration process derives calibration parameters for risk modeling using reference and market.

Among the most important of these inputs is the Variance-Covariance matrix, which is critical for both Parametric VAR/ETL as well as Monte Carlo based VAR/ETL. Others include the yield curves for risk free and risky interest rates, or the Beta for a given firm or equity instrument, or the volatility of a share used as underlying in an equity option.

It makes sense to distinguish those into two categories:

Cashflow Element Mapping Pattern for a Bullet Redemption			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Tag	VARCHAR2	255	
Element Type	**VARCHAR2**	**32**	**"Fixed-Redemption-Bullet-Style"**
Element Analytic Model	VARCHAR2	32	
Element Sequence	NUMBER	22	
Element Operator	VARCHAR2	32	
Element Tag	**VARCHAR2**	**255**	***Redemption Reference***
Element Parent Tag	VARCHAR2	255	
Period Start Date	**DATE**	**7**	***Issue or Accrual Start Date***
Period End Date	**DATE**	**7**	***Redemption Date***
Is Detachable	NUMBER	22	
Base Instrument Identifier Type	**VARCHAR2**	**32**	**"ISO-Currency-Code"**
Base Instrument Identifier	**VARCHAR2**	**32**	***Redemption Currency***
Base Value	NUMBER	22	
Base Value Denominator	NUMBER	22	
Base Unit	VARCHAR2	32	
Trade Instrument	NUMBER		
Trade Instrument Identifier Type	VARCHAR2	32	
Trade Instrument Identifier	VARCHAR2	32	
Trade Value	NUMBER	22	
Trade Unit	VARCHAR2	32	
Rate Constant	**NUMBER**	**22**	***Redemption Price***
Rate Factor	NUMBER	22	
Cap	NUMBER	22	
Floor	NUMBER	22	
Minimum Step	NUMBER	22	
Maximum Step	NUMBER	22	
Amortisation Factor	NUMBER	22	
Payout Factor	NUMBER	22	
Is Accumulating	NUMBER	22	
Underlying Instrument	NUMBER		
Underlying Instrument Identifier Type	VARCHAR2	32	
Underlying Instrument Identifier	VARCHAR2	32	
Underlying Type	VARCHAR2	32	
Underlying Price Type	VARCHAR2	32	
Index Factor	NUMBER	22	
Index Base	NUMBER	22	
Upper Bound	NUMBER	22	
Lower Bound	NUMBER	22	
Exercise Style	VARCHAR2	32	
Mandatory Exercise	NUMBER	22	
Date Accrual Style	VARCHAR2	32	
Date Accrual Rule	**VARCHAR2**	**32**	***Accrual Rule***
Settlement Style	VARCHAR2	32	
Instrument Build	VARCHAR2	32	
Maturity Period Amount	NUMBER	22	
Protected Obligation	VARCHAR2	32	
Credit Protection	VARCHAR2	32	
Right Owner	VARCHAR2	32	
Paying Party	NUMBER	22	

Figure 7.11: Mapping a bullet redemption to the cash flow element.

Cashflow Element Mapping Pattern for a Floating Coupon			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Tag	VARCHAR2	255	
Element Type	**VARCHAR2**	**32**	**"Floating-Rate-Coupon"**
Element Analytic Model	VARCHAR2	32	
Element Sequence	NUMBER	22	
Element Operator	VARCHAR2	32	
Element Tag	**VARCHAR2**	**255**	***Coupon Reference***
Element Parent Tag	VARCHAR2	255	
Period Start Date	**DATE**	**7**	***Accrual Start Date***
Period End Date	**DATE**	**7**	***Last Coupon Date***
Is Detachable	NUMBER	22	
Base Instrument Identifier Type	**VARCHAR2**	**32**	**"ISO-Currency-Code"**
Base Instrument Identifier	**VARCHAR2**	**32**	***Redemption Currency***
Base Value	NUMBER	22	
Base Value Denominator	NUMBER	22	
Base Unit	VARCHAR2	32	
Trade Instrument	NUMBER		
Trade Instrument Identifier Type	VARCHAR2	32	
Trade Instrument Identifier	VARCHAR2	32	
Trade Value	NUMBER	22	
Trade Unit	VARCHAR2	32	
Rate Constant	**NUMBER**	**22**	***Coupon Spread***
Rate Factor	**NUMBER**	**22**	***Multiplier-for-Underlying-Rate***
Cap	**NUMBER**	**22**	***Rate Cap***
Floor	**NUMBER**	**22**	***Rate Floor***
Minimum Step	NUMBER	22	
Maximum Step	NUMBER	22	
Amortisation Factor	NUMBER	22	
Payout Factor	NUMBER	22	
Is Accumulating	NUMBER	22	
Underlying Instrument Identifier Type	**VARCHAR2**	**32**	***Rate-Identifier-Type***
Underlying Instrument Identifier	**VARCHAR2**	**32**	***Rate-Identifier***
Underlying Type	**VARCHAR2**	**32**	**"Underlying-Reference-Rate"**
Underlying Price Type	**VARCHAR2**	**32**	***Price Type***
Index Factor	NUMBER	22	
Index Base	NUMBER	22	
Upper Bound	NUMBER	22	
Lower Bound	NUMBER	22	
Exercise Style	VARCHAR2	32	
Mandatory Exercise	NUMBER	22	
Date Accrual Style	VARCHAR2	32	
Date Accrual Rule	**VARCHAR2**	**32**	***Accrual Rule***
Settlement Style	VARCHAR2	32	
Instrument Build	VARCHAR2	32	
Maturity Period Amount	NUMBER	22	
Protected Obligation	VARCHAR2	32	
Credit Protection	VARCHAR2	32	
Right Owner	VARCHAR2	32	
Paying Party	NUMBER	22	

Figure 7.12: Mapping a floating rate coupon to the cash flow element structure.

Cashflow Element Mapping Pattern for an Equity Call Option			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Tag	VARCHAR2	255	
Element Type	**VARCHAR2**	**32**	**"Equity Call Option"**
Element Analytic Model	VARCHAR2	32	
Element Sequence	NUMBER	22	
Element Operator	VARCHAR2	32	
Element Tag	**VARCHAR2**	**255**	***Option Reference***
Element Parent Tag	VARCHAR2	255	
Period Start Date	**DATE**	**7**	***Option Start Date***
Period End Date	**DATE**	**7**	***Last Exercise Date***
Is Detachable	NUMBER	22	
Base Instrument Identifier Type	**VARCHAR2**	**32**	**"ISO-Currency-Code"**
Base Instrument Identifier	**VARCHAR2**	**32**	***Option Currency***
Base Value	NUMBER	22	
Base Value Denominator	NUMBER	22	
Base Unit	VARCHAR2	32	
Trade Instrument	NUMBER		
Trade Instrument Identifier Type	VARCHAR2	32	
Trade Instrument Identifier	VARCHAR2	32	
Trade Value	NUMBER	22	
Trade Unit	VARCHAR2	32	
Rate Constant	**NUMBER**	**22**	***Strike Price***
Rate Factor	NUMBER	22	
Cap	NUMBER	22	
Floor	NUMBER	22	
Minimum Step	NUMBER	22	
Maximum Step	NUMBER	22	
Amortisation Factor	NUMBER	22	
Payout Factor	NUMBER	22	
Is Accumulating	NUMBER	22	
Underlying Instrument Identifier Type	**VARCHAR2**	**32**	***Underlying-Identifier-Type***
Underlying Instrument Identifier	**VARCHAR2**	**32**	***Underlying-Instrument-Identifier***
Underlying Type	**VARCHAR2**	**32**	**"Underlying-Instrument"**
Underlying Price Type	**VARCHAR2**	**32**	***Price Type***
Index Factor	NUMBER	22	
Index Base	NUMBER	22	
Upper Bound	NUMBER	22	
Lower Bound	NUMBER	22	
Exercise Style	**VARCHAR2**	***32***	**Exercise Style (e.g., European or American)**
Mandatory Exercise	**NUMBER**	***22***	**Mandatory Exercise**
Date Accrual Style	VARCHAR2	32	
Date Accrual Rule	VARCHAR2	32	
Settlement Style	**VARCHAR2**	***32***	**Settlement Style (e.g., Cash or Physical)**
Instrument Build	VARCHAR2	32	
Maturity Period Amount	NUMBER	22	
Protected Obligation	VARCHAR2	32	
Credit Protection	VARCHAR2	32	
Right Owner	VARCHAR2	32	
Paying Party	NUMBER	22	

Figure 7.13: Mapping an equity option to the cash flow element structure.

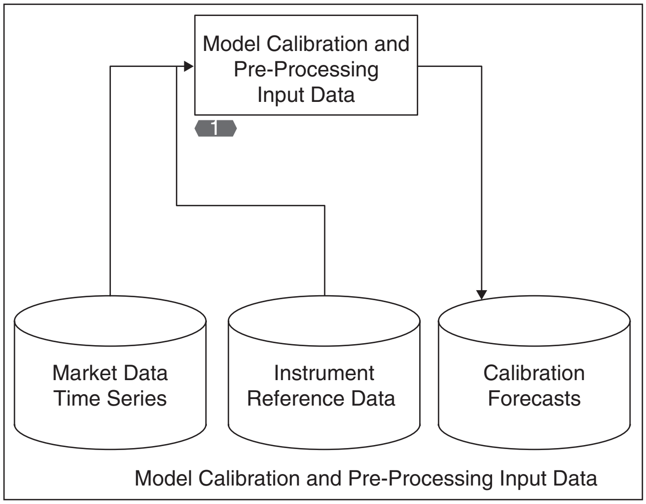


Figure 7.14: Model calibration and pre-processing input data.

1. the calculation of constructs like yield curves, which can be called model calibration (or just calibration) since a yield curve and similar constructs provide generic parameters for models not directly related to an instrument to be valued.
2. the task of pre-processing input data such as a time series of market prices of an instrument, some index or time series for related dividends or earnings, and other data into a forecast for the price volatility of the instrument, its Beta relative to the benchmark, the growth rate of the dividend or earnings stream over time.

In some cases this pre-processing can be fully automated, but in others it will involve considerable human input and intervention. Measuring the *historical* Beta or Volatility can be fully automated. Volatility and Beta *forecasts* can sometimes be automated but at least need to be monitored closely. Others like earning or even dividend forecasts usually need substantial input and intervention from analysts.

In your implementation you need to determine which types of calibration and pre-processed inputs you need. Then you need to determine how they can best be mapped into your modelling database. After that you need to decide which type of process will create the calibration and pre-processed inputs, whether this process is part of your implementation, or external to it. If the process is external to your implementation you need to decide how you can best feed them into your database, and whether these feeds can be fully or partly automated. You can use Figure 7.15 as starting point for a checklist to guide you through the process of deciding what calibration processes you may need and how to realize them.

If any calibration and pre-processed input data is part of your project and cannot be obtained from another source, you will then need to decide if their creation can be fully or

Calibration and Pre-Processed Inputs

- Which Calibration and Pre-Processed Inputs are needed?
- How can they be mapped into the database?
- Which of them are part of your implementation and which not?
- How can you feed those that are external and can this be automated?
- Which of those that are not external should be fully or partly automated?

Figure 7.15: Checklist for implementing calibration and pre-processed input data.

partially automated. Otherwise, it remains a manual activity for the user of your pricing models. If any can be automated you will need to ensure that they fit into your overall processes. Where necessary, they need to be supported by tools such as input (or operator) screens, parameter files so that your users or model operators can adjust the processes to changing circumstances.

Lab Exercise 7.2: Exploring Risk Calibration and Pre-processed Input Data

1. Download the tutorial notes **LAB_7_2_ExploringRiskCalibrationData.pdf** from the companion web site (http://modelbook.bancstreet.com/) to explore how calibration and pre-processed input data like Variance-Covariance matrices can be created and stored in the database and common issues that arise from creating such data or using external feeds.

7.1.4 Creating Risk Measurement Processes

Once your reference and market data is in your database, and you have correctly set up your exposure aggregation hierarchy, mapped all building blocks to pricing models and implemented those risk analysis models, and completed any work for automating feeds or the creation of calibration and preproceessed input data, you will still need to orchestrate the overall risk measurement calculation process.

The end-to-end process usually comprises many subprocesses, all of which need to be coordinated. Figure 7.16 illustrates a typical end-to-end process in three stages. Although it is possible to do this coordination across subprocesses manually, more often than not this

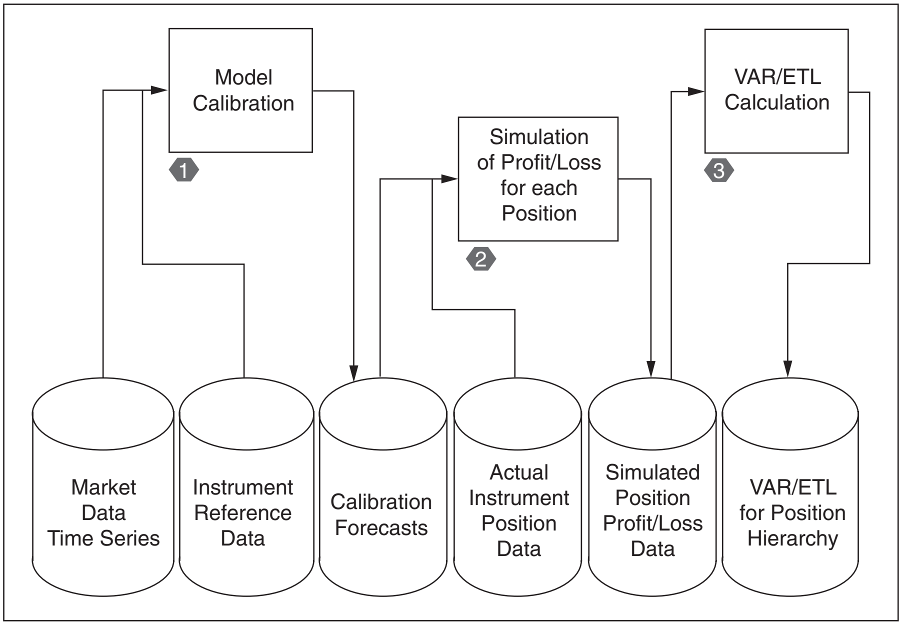


Figure 7.16: End-to-end risk measurement process.

would be very risky and costly in terms of required manpower. If this is the case you need to decide how you can implement the orchestration for your valuation solution with the resources you have at hand. In some cases it is possible to write a simple program component or even just a script that ties all your processes together.

Usually, however, you need a more flexible solution than that. One of the big benefits of the building block approach is its flexibility and the ease and speed with which you can adapt to new financial products or new approaches to pricing particular atomic building blocks. This benefit would be lost if you implement your solution as one monolithic block so that your whole solution has to be reworked, retested, and reinstalled every time you want to change your high-level end-to-end process that runs your valuation models or any time you want to add or modify a single building block valuation model.

One way to avoid this predicament is to use a meta model for defining your high-level end-to-end process and then either create some software yourself that lets you execute the process as defined by the meta model or to use a readymade solution to both define and run your process.

Before you jump into building such a solution yourself make sure you are both an expert in executable process models, the implementation of process schedulers, and a glutton for extreme punishment. The smarter way to implement your end-to-end process is to use a commercial ready-to-use solution for this. The easiest, quickest, and often the most robust solution specifically designed for the implementation of end-to-end pricing are risk and performance measurement processes such as Fincore Financial's Analytics Hub (www.fincorefinacial.com).

Alternatively you could use a generic solution such as the ORACLE business process orchestration toolkit, or you could use a process scheduler your organisation already uses together with any middleware you may need to link up your models, the scheduler, and your database. Although it is not as bad as building your own from scratch, this alternative still involves a substantial amount of effort, risk, and costs, and you should think very carefully whether using a generic process orchestration toolkit or an existing general purpose scheduler and some glue logic will be meeting your needs.

If you use a solution specifically designed for the implementation of end-to-end risk measurement processes such as Fincore Financial's Analytics Hub, you will be able to focus on designing your models and process, and will not have to spend time creating the infrastructure they need to operate. Figures 7.17 and 7.18 illustrate how easy it can be to set up your end-to-end process. Figure 7.18 shows how you could register an existing or new pricing model and set up its parameters.

Such parameters can be either values or queries you can enter and save in a different screen that select the data according to your given criteria when the end-to-end process is run. Such user-defined queries give you full control over the process and are very powerful. A single risk model could be set up many times, each time with different parameters. Each

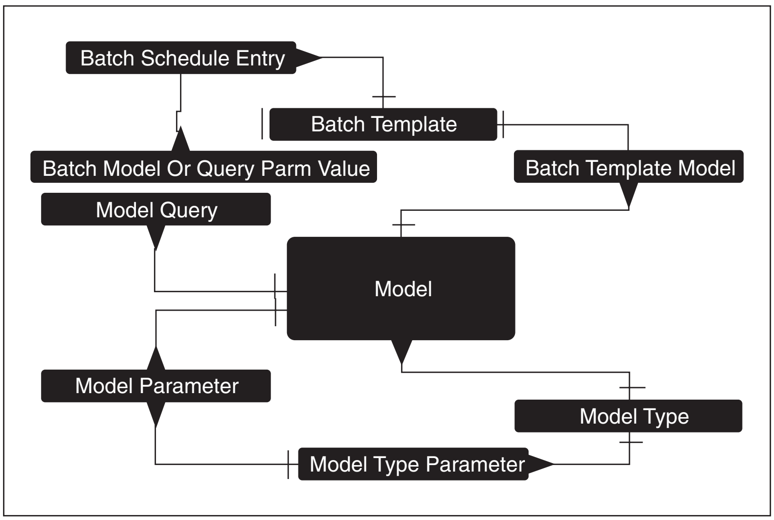


Figure 7.17: A meta model for your high level valuations process.

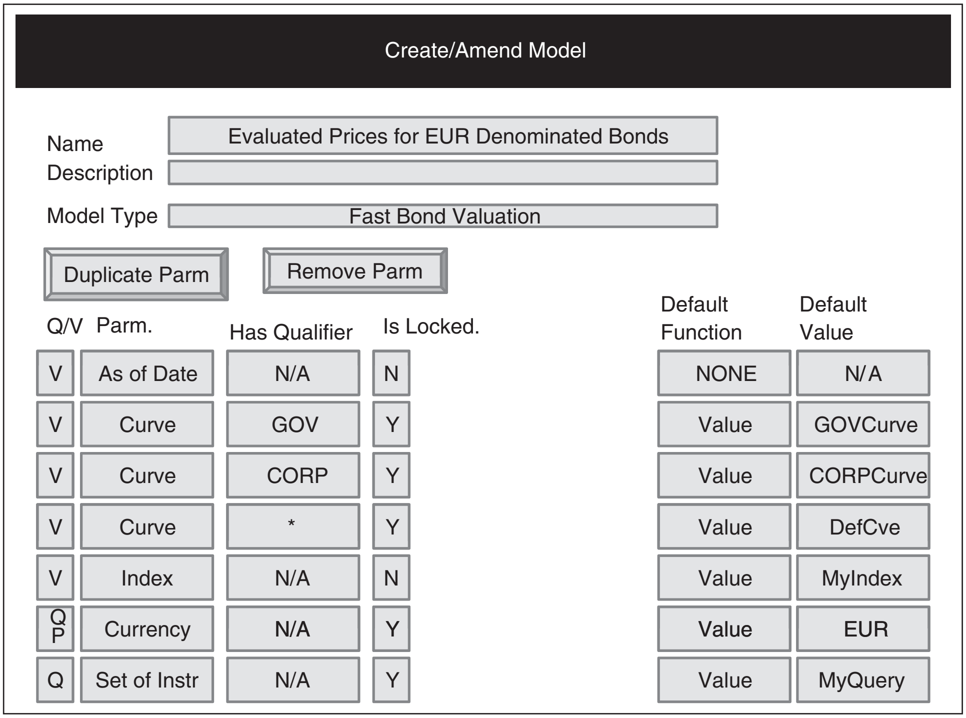


Figure 7.18: Registering valuation model components as part of a process meta model.

such model configuration thus would operate on different instruments and with different calibrations or different versions of input data.

Figure 7.19 shows how you might put together different models into one overall process simply by choosing from your pool of model configurations.

Lab Exercise 7.3: Exploring End-to-End Risk Modelling Process Configurations

1. Download the tutorial notes **LAB_7_3_EndToEndRiskModellingProcesses.pdf** from the companion web site (http://modelbook.bancstreet.com/) to explore how you can define end-to-end valuation processes in a purpose built framework for defining and running valuation processes.

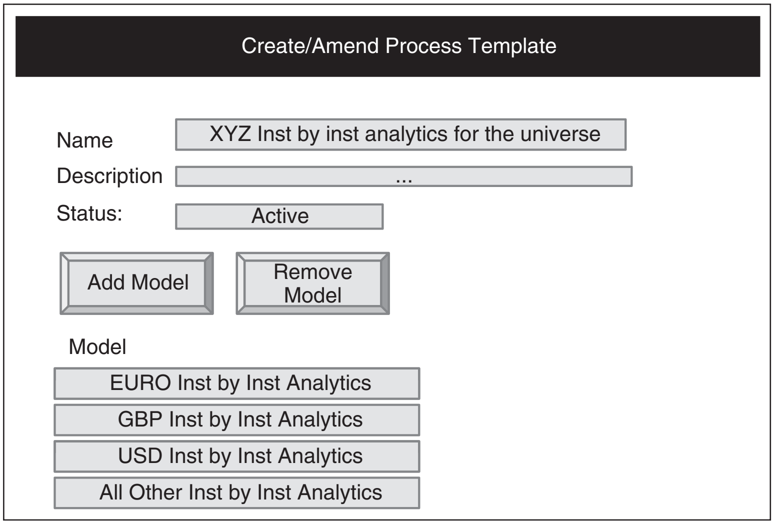


Figure 7.19: Creating a process template ready to be run using a process scheduler.

7.1.5 Refining Your Risk Model Implementation

Risk Model implementation involves large numbers of assumptions and judgement calls that have to be made well before a result becomes visible. It is inevitable that at least some assumptions and judgement calls will have to be revised.

The fewer assumptions and judgement calls are involved and the quicker you can get to the point where you can check them against real results, the easier it will be to identify what needs to be changed and to put the revisions into practice.

In Chapter 1 we introduced the fountain model for model implementation that is reproduced in Figure 7.20. It provides a robust framework that will help you make sure that frequent small iterations do not veer off course and your implementation becomes an exercise in "extreme model hacking."

In practical terms this means that you should start your implementation with one or at most two risk models that are easiest and with which you have most experience. If these models can be mapped to many atomic building blocks, start by covering at most a handful of atomic building blocks from those that can be mapped to the models you are actually implementing. Even for simple historical data-based risk models this will still leave you with a challenging amount of effort. Figure 7.21 gives you a checklist to use as a starting point for keeping your iterations small, manageable, and productive.

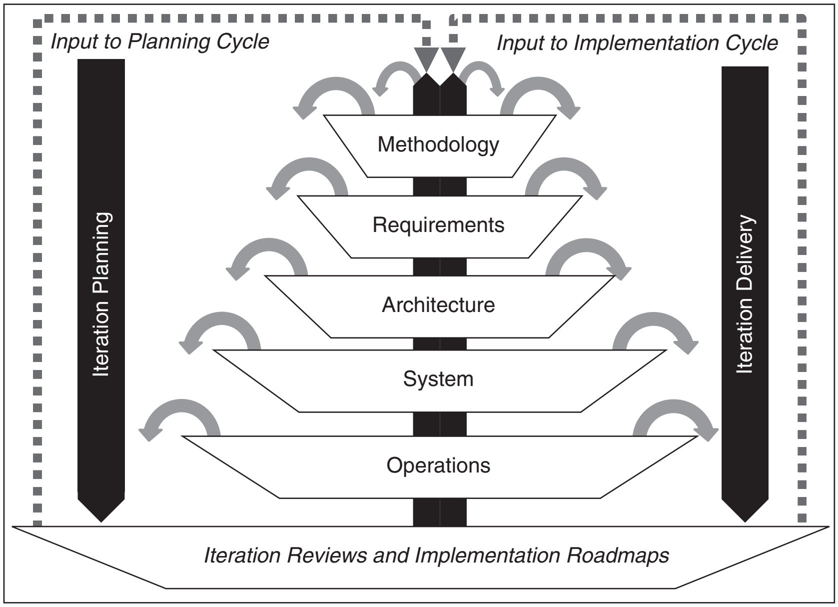


Figure 7.20: Iterative refinement of model implementations.

Small Iterations as the Road to Success

- Start with a single risk model and get it working end-to-end
- Start with the easiest most familiar risk models first
- Use external calibration and pre-processed input data first
- In later iterations limit the number of models for calibrations or pre-processed input data to 1–2 per iteration
- Aim for iterations no longer than one month
- Early iterations should be as short as possible, try one week iterations if at all possible
- Focus on modelling rather than infrastructure – if possible use a purpose designed model implementation platform such as Fincore Financial’s Analytics Hub or other similar solution

Figure 7.21: Small iterations as the road to success.

7.2 Risk Model Implementation Checklists

Figures 7.22 and 7.23 summarise the framework for model implementation that we introduced in Chapter 1. Both charts are meant to act as checklists rather than a rigid program of mandatory tasks and deliverables that have to be complete after each iteration.

You may find that, for instance, the Governance and Oversight Perspective as well as the Business Ownership perspective may receive significant attention in the first one or two iterations so that a suitable foundation for future iterations is put together.

Subsequent iterations will be much lighter in terms of effort dedicated to these two perspectives, since the fruits of this early effort can be used again and again in later iterations. At some point, maybe when you are about to shift gears from early prototypes and pilots to having your solution in full-scale production, the Governance and Oversight Perspective as well as the Business Ownership perspective are likely to come back into the foreground. The nature of your work is then shifting again and you will need to revise the ground rules to which you had agreed earlier with your stakeholders and business sponsors.

When you start planning for an iteration the two charts are ideal tools for deciding on which aspects you need to focus on. Just like a pilot and co-pilot in charge of a plane go through their pre-flight checklist prior to take-off, you and your team should go through the cells in each diagram and make a call of whether the item in the cell needs attention in the forthcoming iteration. Then, your planning and work can concentrate on those items that matter for the current iteration.

7.2.1 Methodology Perspective: Risk Governance Oversight

In this perspective you take a town planner's view of your risk implementation. What is important is that you make the ground rules to fit the size and nature of your endeavour. If you simply tinker with risk models on the side, perhaps to explore their potential or better understand them, then you will need very little groundwork in this perspective.

If, on the other hand, your risk models are used for key financial decision making or will be used as part of a service or product then the work that has to be undertaken in this perspective will be substantial. The key points your planning should cover are:

- Who will use your risk models and for what purpose will they be used?
- What will happen if your risk models are wrong or unavailable?
- Who would have to answer the questions if things go wrong after this or the next few iterations: you or your team, your boss, the CEO or the Chairman of the Board of your organisation?

Model Implementation Framework							
Perspective	Goals Dimension	Organisation Dimension	Schedule Dimension	Model Dimension	Data Dimension	Process Dimension	Role
Methodology Perspective	Goal Setting Guidelines	Organisation Guidelines	Schedule Guidelines	Modelling Guidelines	Data Guidelines	Process Guidelines	Business Governance
Business Perspective	Business Case/Plan	Organisation Requirements	Schedule Requirements	Model Requirements	Data Requirements	Process Requirements	Business Owner
Architecture Perspective	Architecture Plan	Organisation Specifications	Design Schedule	Model Specifications	Data Specifications	Process Specifications	Model Designer
Systems Perspective	Build Plan	Responsibility Model	Build Schedule	Model Designs	Msg & Data Store Design	Process Designs	Model Builder
Operating Perspective	Operating Plan & Targets	Responsibility Assignments	Maintenance Schedule	Production Models	Production Data Stores	Operating Schedules	Model Operator
	Why?	Who?	When?	How?	What?	Where?	

Figure 7.22: Model implementation framework.

Model Implementation Methodology Template							
Set of Deliverables	Goals	Organisation	Schedule	Model	Data	Process	Persons Responsible
Methodology	Develop and implement goal setting guidelines	Develop and implement organisation guidelines	Develop and implement scheduling guidelines	Develop and implement modelling guidelines	Develop and implement data guidelines	Develop and implement process guidelines	Governance Board
Requirements	Develop and implement business case & plan	Develop and implement organisation requirements	Develop and implement schedule requirements	Develop and implement model requirements	Develop and implement data requirements	Develop and implement process requirements	Business Owner
Architecture	Develop and implement architecture plan	Develop and implement organisation specifications	Develop and implement design schedule	Develop and implement model specifications	Develop and implement data specifications	Develop and implement process specifications	Model Designer
System	Develop and implement build plan	Develop and implement responsibility model	Develop and implement build schedule	Develop and implement model designs	Develop and implement msg & data store design	Develop and implement process designs	Model Builder
Operations	Develop and implement operating plan & targets	Develop and implement responsibility assignments	Develop and implement maintenance schedule	Develop and implement production models	Develop and implement production data stores	Develop and implement operating schedules	Model Operator
	Why?	Who?	When?	How?	What?	Where?	

Figure 7.23: Model implementation methodology template.

- Have you included the right people in the dialogues or workshops for setting out the ground rules?
- Have you done enough work to make the ground rules robust enough for the current stage?
- Have you we covered all the different dimensions from Goals to Process?
- Are the ground rules just right: easy to understand, easy to put into action, easy to measure but complete, consistent, and comprehensive enough to rule out pitfalls into which you or your team, your division, or even your whole firm might fall?

7.2.2 Business Perspective: Risk Business Ownership

In this perspective you take the view of a business owner. Two questions are always (or at least always should be) on a business owner's mind:

- How will this make us more profitable?
- How will this protect your business from failing?

Thus you will need to probe for how your risk measures and the way you create them add value to your business, to that of your sponsors or that of your clients. Yet again, check how your risk models—when they go wrong—could endanger your business or that of your sponsors or clients. The key points your planning should cover are:

- Are the risk models and process for creating them fit for the intended purpose?
- Does your proposed way of creating risk models fully exploit the potential for adding value to your business or that of our sponsors or clients?
- Can you do this more cheaply, faster, with less resources, fewer errors, or in a more informative way?
- What is the biggest hole you can fall into with the chosen approach to risk modelling and how would you get out of it? Would you survive the fall?
- Have you got the resources to see our plan through to completion? Can you see it through beyond completion and all the way until it will be a success?

7.2.3 Architecture Perspective: Risk Model Designer

The designer's—or architect's—perspective is dealing the decisions that set the framework for how your models will be implemented in detail. Some of these decisions may be given quantities if you already have a technical infrastructure for your implementation that you must use.

The challenge for the designer is to shape the remaining points in the framework in such a way that both the model builder's work and the model operator's efforts will together yield the desired results both now and in the distant future. The key points your planning should cover are:

- What is the right infrastructure for your implementation?
- Do you have the right infrastructure, and if not, how can you get it?
- Will your proposed implementation work with your given data feeds and the skills and experience of the people who will operate the models?
- What is the road map that will deliver as much value as possible as early as possible?
- Are you using this roadmap? If not, what stops you from using it and how can you remove and such blocks?

7.2.4 Systems Perspective: Risk Model Builder

The system builder's perspective deals with the shop floor decision of how to actually code and put together the risk model solution. Here, you can rely on a lot of ground work already laid by the other perspectives. The bigger picture in terms of what you aim to achieve, the pitfalls to avoid, and the constraints given or agreed should already be clear. Thus, you can now focus on the details of the implementation. The key points your planning should cover are:

- Are you implementing the model in the best way possible?
- Are there faster or more predictable ways at arriving at the same result?
- Are you handling all the pitfalls that your valuation model may face?
- Can and do you double-check the results? If not is this OK under the circumstances? If the answer is yes, would your boss, the CEO or your clients or regulators agree? If they would agree now would they still agree if our risk measurements went horribly wrong and undetected for some time?
- If you detect a problem later or a better method becomes available, how difficult would it be to switch? If it is difficult, what can you do now to make that switch easier?

7.2.5 Operating Perspective: Risk Model Operator

The operator's perspective is dealing with what it is like to use the risk measurement solution to produce actual VAR or ETL measures under actual operating conditions rather than what may have been assumed or planned. The key points your planning should cover are:

- Do you fully understand the capabilities and limitations of the solution? If not, where do you lack information and how could you bridge this gap?
- Can you run the solution in such a way as to produce valuations to an agreed standard with the given data feeds and the given expertise of the team?
- If not what are the problems and what can be done in the short term to overcome them?
- How can you help the implementation team to overcome these challenges in the medium to long term?
- Are you making the best use of the solution? If not what would you need to do to get there?
- What could go wrong with the way you operate the solution? What is the impact on the business and our clients in the worst case? Are there ways of mitigating this? Are there ways in which the solution could be operated differently that would result in a more favourable worst case or at least reduce the impact?

7.3 Further Risk Models and Implementation Details from Historical to Parametric and Monte Carlo Based Approaches

In Chapter 6 we introduced a basic set of commonly used risk measurement models covering debt instruments, equity, and even simple options. There are of course many more in which you may be interested.

Although there is no hope of covering everything even in a book just on risk models, the more models to which you have access, the more you will be able to practice. We therefore have put together several lab exercises that will introduce you to more risk measurement models that we could not fit into this book.

Lab Exercise 7.4: A More Advanced Historical VAR/ETL Model

1. Download the tutorial notes **LAB_7_4_AdvancedHistoricalVAR/ETLModel.pdf** from the companion web site (http://modelbook.bancstreet.com/) to explore a more advanced historical VAR/ETL model and its implementation.

Lab Exercise 7.5: A Parametric VAR/ETL Model and Its implementation

1. Download the tutorial notes **LAB_7_5_ParametricVARETLModel.pdf** from the companion web site to explore a Parametric VAR/ETL model and its implementation.

Lab Exercise 7.6: A Monte Carlo VAR/ETL Model and Its Implementation

1. Download the tutorial notes **LAB_7_6_MonteCarloVARETLModel.pdf** from the companion web site to explore a Monte Carlo VAR/ETL model and its implementation.

This page intentionally left blank

CHAPTER 8

Introducing Performance Measurement

In this chapter, we explain the role of performance measurement in the investment process and for the management of financial institutions in general. We demonstrate how to

- calculate money- and time-weighted performance measures for assessing overall portfolio performance
- define customised equity style benchmarks
- calculate and use classic risk adjusted performance measures for comparing portfolios
- perform attribution analysis using classic equity portfolio attribution tools
- calculate and use RAROC risk adjusted return on capital measures to compare performance at a company level

Finally, we explain the basic portfolio optimisation process and its place in the investment performance cycle.

8.1 The Role of Performance Measurement

Performance measurement is at the very heart of the investment process, and indeed, of basically every financial services business. The reason for this is that performance (including risk criteria) is both the starting point for every investor and ultimately the only result that truly matters. In most other industries, the product or service delivers a broad range of benefits to a customer, and pure functional performance is often only one of several key benefits. In the investment industry this is not normally the case: Ancillary services can do very little against problems a portfolio manager may have when delivering the required target performance (including risk criteria).

In Figure 8.1 you can again see the key activities that make up the investment process. In the box with the title Performance and Risk Management there four activities. Two of them, Portfolio Valuation and Risk measurement, we looked at in earlier chapters. The bulk of this chapter and the next one will focus on the details of the remaining two: Performance Measurement and Performance Attribution. Performance Measurement is a subtask of the larger set of performance and risk management activities and is concerned with accurately

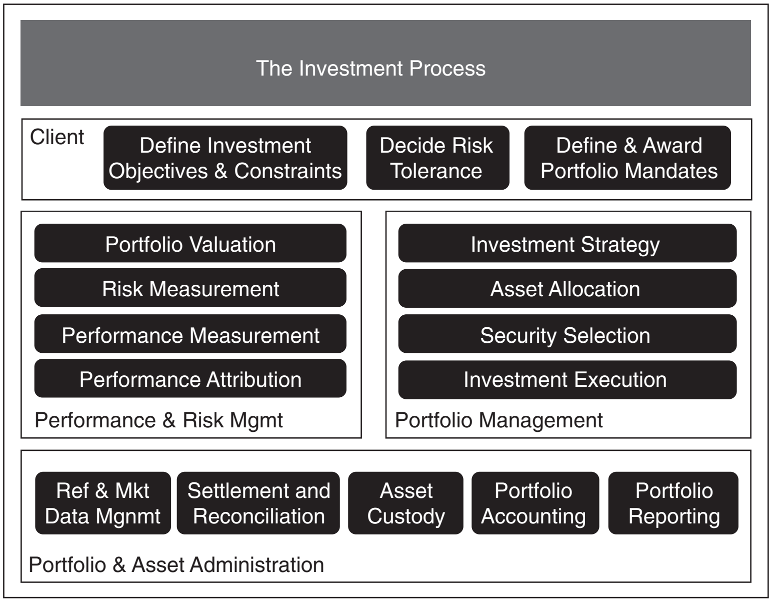


Figure 8.1: The investment process.

measuring and communicating the true results achieved by an investment manager with specific portfolios. Performance Attribution then further analyses the portfolio performance results to determine the reason or source of an out- or underperformance.

8.1.1 Investment Management and Other Businesses

In this book we look at performance mainly from an asset manager's perspective. We will explore performance measurement in the wider context of other types of financial services firms like banks and insurance companies when we look at risk adjusted returns on capital. Asset management, in the way we look at it, is not limited to traditional pension and investment fund management or private client portfolio management. It also occurs in other areas of banks and insurance companies managing different assets such as trade and credit portfolios or portfolios of real estate. The key is that in each case a portfolio manager is given the mandate to manage a given amount of the investor's capital that he or she will use to invest in a defined set of assets with a view of obtaining returns that are in line with the investor's risk tolerance and a reasonable performance benchmark pre-agreed upon.

We will look in more detail at role of performance measurement in the relationship between the portfolio manager and the investor in the following section on governance and accountability. Performance measurement, including portfolio valuation, risk and performance measurement in a narrow sense as well as performance attribution, is very important for the portfolio manager or portfolio management team. In Figure 8.2 we

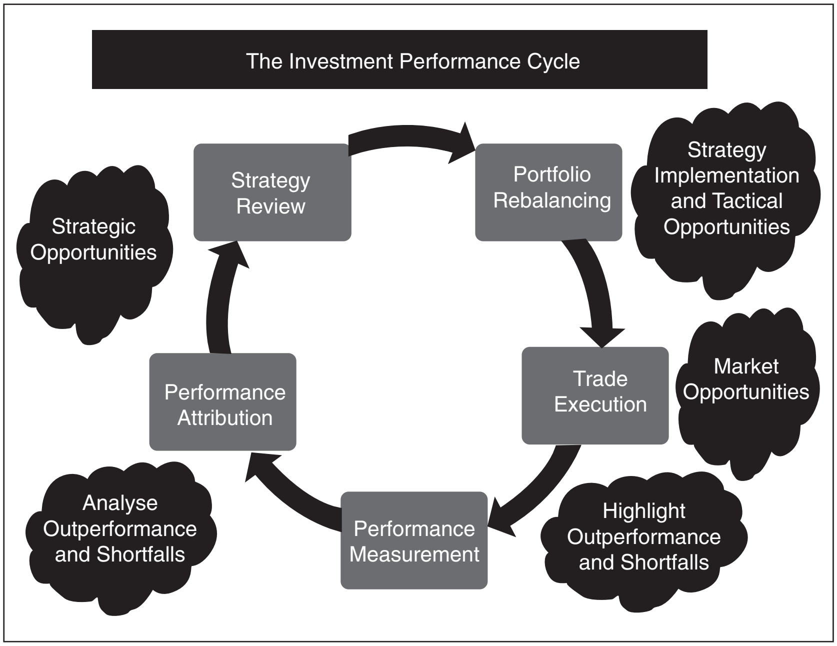


Figure 8.2: The Investment Performance cycle.

consequently show the role of performance measurement in the investment performance cycle. Portfolio management is not a one-off task, but rather a highly iterative endeavour based on a continuous cycle of activities.

The results of analysing outperformance and shortfalls during performance attribution feed directly into the ongoing portfolio strategy review activity and are an important input to the resulting investment decisions. Performance attribution results are also taken very seriously in the next step of portfolio rebalancing. Segments or individual stocks that have made a negative contribution to the performance can receive particular scrutiny and if warranted be eliminated from the portfolio or receive a reduced weight. Outperforming segments and stocks too can be similarly scrutinised to verify that their strong performance is likely to continue, in which case their weight may be kept as is or even be increased. If an outperforming segment or stock has had its day, facing difficulties with falling or even negative return, its weight is again likely to be reduced or eliminated all together.

In the following step, of trade execution, the results of a detailed trade level performance analysis, can help build up the market knowledge that is required to reduce and keep the trading costs for the portfolio as low as possible. The market impact of careless trade execution when implementing portfolio decisions can seriously reduce a portfolio's performance, especially if a portfolio is substantial or contains any instruments for which the market is less than perfectly liquid, deep, and continuous.

After trade execution, the performance cycle restarts again with performance measurement and performance attribution. It measures the performance of the portfolio both in the light of the evolution of markets and if there have been any changes to the portfolio in the light of those changes. Although-historically-the measurement cycle has been rather lengthy, ideally this cycle should be a daily one. With a daily performance cycle much useful information can be fed back directly into the investment process. This helps to avoid small problems from becoming big enough to have a serious impact, but it also can highlight opportunities that might otherwise have been spotted only later or not at all.

8.1.2 Governance and Accountability

The investor who has delegated the management of a portfolio to a portfolio manager will be interested not just in the pure risk aspects of the portfolio that we explored in Chapters 6 and 7, but also in their performance more generally. For the protection of private investors, many countries have set down minimum standards for content and reporting cycles. In fact, most investment managers substantially exceed these minimum standards, both in terms of content and the frequency of reports.

Institutional investors demand highly detailed and frequent performance information. Monthly performance reports, which include performance summaries like the **Example Performance Summary Report** shown in Figure 8.3 as well as large amounts of detailed supplementing information are usually considered as an absolute minimum by institutional investors. Most professional investors want specialised and ad hoc reports, or access to detailed performance data that they can analyse themselves to construct their own tailored reports.

Accurate, detailed, and timely performance information is an important incentive for portfolio managers to discharge their responsibility toward the investor with the very highest level of professionalism. Such best practice performance reporting will make it less likely for the portfolio manager to be penalised for factors out of his or her control that impact the portfolio returns. Good performance due to skill and diligence will be more easily differentiated from sheer good luck or simply running with the herd in a bull or bubble market.

The investor will be less tempted to take flight and compound any losses in the event of temporary hits to performance by unusual transitory circumstances outside the control of the portfolio manager. On the other hand, if the manager's ability or willingness to discharge their mandate is not measuring up to the investor's requirements, the existence of accurate, detailed, and timely performance information will help to spot any problems early on. This will allow to either resolve them or terminate the mandate before too much damage is done to the investor's capital.

bancstreet
capital partners

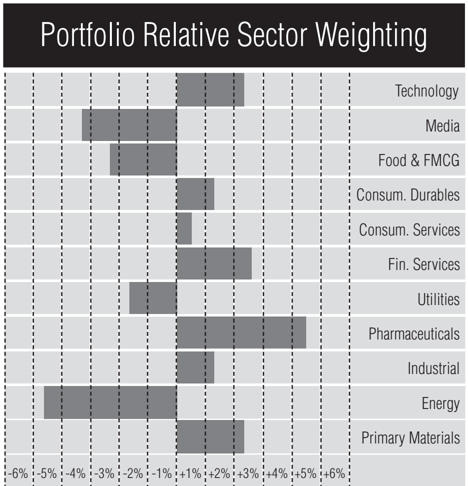
Portfolio Performance Report
Client XYZ - Sample Portfolio

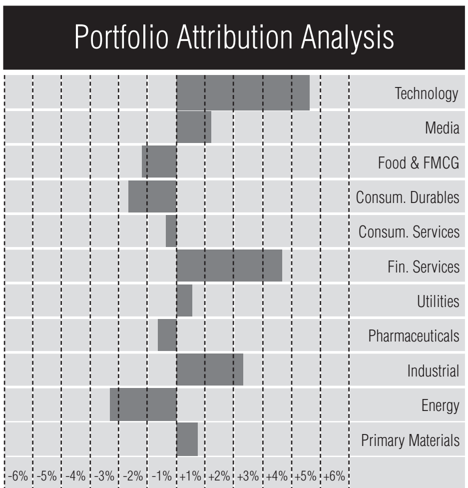
Performance and Risk Summary

Analysis by Factor

Client: XYZ General Insurance Plc
Period: 01.01.2010 to 31.01.2010
Ref. Currency: GBP

Portfolio ID: XYZ-GEN-EQU-BAL
Client-Manager: Sally Sample-Smith
Benchmark: XYZ-GEN-ALM-BMK





Return Analysis	
Portfolio	+0.34%
Benchmark	–3.86%
Net Active Return	+1.52%

Attribution Analysis by Sector	
Asset Allocation	+0.38%
Stock Selection	+0.27%
Interaction Effect	+0.11%
TOTAL	+0.76%

Attribution Analysis by Region	
Asset Allocation	+0.38%
Stock Selection	+0.27%
Interaction Effect	+0.11%
TOTAL	+0.76%

Risk Analysis	*Ptf*	*Bmk*
No of Instruments	69	382
No of Currencies	3	4
Portfolio Value	46,344,456 GBP	
ETL at 99%, 30 days	1,123,400 GBP	
VAR at 99%, 30 days	851,130 GBP	
Tracking Error (ex-post)	1.95	
Information Ratio	1.95	
Sharpe Ratio	1.5	1.5
Treynor Ratio	1.95	
Total Risk Alpha	1.5	1.5
Jensen Alpha	1.95	
RAP – Total Risk	1.5	1.5
RAP – Market Risk	1.95	
DR – Total Risk	1.5	1.5
DR – Market Risk	1.95	

Figure 8.3: Example performance summary report.

The very first time an investor will get to see detailed performance information is usually during the selection process. Any question marks at this early stage over the accuracy, level of detail, timeliness, or the consistency of the performance data with the investor's own information about the performance of markets or investments should mean the removal of the manager from the selection shortlist.

8.1.3 Performance Management

Many performance and attribution measures were designed for individual portfolios or the comparison of a small set of portfolios. In some circumstances this is not enough. Consider for instance an insurance company or a bank with very different portfolios and some business activities that, although not actually portfolios, should be compared with portfolios. The need that drives such performance comparison is usually the need to allocate a firm's capital efficiently between different investment and trading portfolios as well as other activities. In this and the following chapter we will therefore also take a look at Risk Adjusted Return on Capital (RAROC) measures that help to implement performance management in a wider context of firms or other organisations where the classic portfolio-based measures are not universally applicable. Although RAROC is a relatively new concept in a portfolio context it is well established and RAROC measures can be easily derived from accurate, detailed, and timely performance or risk information that is normally readily available in a portfolio context.

8.1.4 Integrated Investment Management

The final part of this chapter will deal with portfolio optimisation. You may be somewhat surprised at this given the title of the chapter. There is a good reason for this, however. In Figure 8.2 we have depicted the investment performance cycle, which includes all the different steps in an iterative performance-driven investment process. We will not expand much on the details of trade execution or investment strategy definition and implementation. Portfolio optimisation, however is part of portfolio rebalancing, and is very closely linked to the risk and performance measurement processes of the preceding chapters and the following one. A suitable robust and efficient process for optimising portfolio composition can make a major contribution to relative performance, minimise the risks for a given level of return, and in particular reduce the emergence of unexpected losses outside the investor's risk tolerance zone.

8.2 Performance Measurement

Performance Measurement is the basis on which all other investment analysis such as performance attribution or performance management are based. It answers two key questions for both investors and investment managers: Where are we, and how well are we doing? The starting point for performance measurement is the valuation of all instrument positions in a portfolio or set of portfolios over time. The next step is measuring the simple performance or return between two valuations for a portfolio. Since many funds and portfolios have a life that spans many years or even decades, we usually want to measure performance over a long time frame. To do this we need to go beyond simple one-period performance measures. The Internal Rate of Return or Money Weighted Performance is a well-known technique from bond analysis, but a measure rather avoided for performance measurement. Although requiring more effort on the part of the performance measurer, Time Weighted Performance including simplified variants such as the Modified Dietz Method are the tools of choice for measuring performance across longer time intervals. We will now briefly look at each of these steps in turn.

8.2.1 Valuation for Performance Measurement

The calculations involved in valuation are as simple as they can get. The value of any position in a portfolio is simply its size in terms of number of instruments times the appropriate market price:

$$\text{Position Valuation} = \text{Position} \cdot \text{Valuation Price}$$

After valuing each position, a number of other relevant measures can be calculated that are both useful on their own account and vital as input for other performance measurement and attribution calculations. From an initial and final value for a period at hand for example you can calculate the unrealised gain or loss for the period as follows:

$$\text{Unrealised Gain or Loss} = \text{Final Value} - \text{Initial Value}$$

Once you have determined the unrealised gain or loss you can calculate the period return for each position as:

$$\text{Period Return} = \text{Unrealised Gain or Loss}/\text{Initial Value}$$

Another statistic that is both useful in reports and needed for later calculations is the weight of each position in proportion to the portfolio value at a given point in time. In reports the weight shown is usually the weight of the position value at the beginning of the period relative to the total portfolio value at the beginning of the period:

$$\text{Position Weight} = \text{Position Value}/\text{Value of Entire Portfolio}$$

The report extract in Figure 8.4 is a typical example of a portfolio valuation report and shows the valuation and other measures set out earlier.

bancstreet
capital partners

Portfolio Performance Report
Client XYZ - Sample Portfolio

Portfolio Valuation Report

All Sectors

Client: XYZ General Insurance Plc
Period: 01/01/2010 to 31/01/2010
Ref. Currency: GBP
Portfolio ID: XYZ-GEN-EQU-BAL
Client-Manager: Sally Sample-Smith
Benchmark: XYZ-GEN-ALM-BMK

Portfolio Valuation by Industry

Name	Position	Initial Price	Final Price	Initial Value	Final Value	Gain/Loss	Return	Weight
				50,000,000	48,002,706	–1,997,294	–3.99%	100.00%
Oil & Gas								
ENI	146,199	17.10	15.92	2,500,000	2,328,097	–171,903	–6.88%	5.00%
REPSOL YPF	35,740	13.99	14.23	500,000	508,739	8,739	1.75%	1.00%
TOTAL	244,349	40.93	39.00	10,000,000	9,530,270	–469,730	–4.70%	20.00%
Basic Materials								
AIR LIQUIDE	8,529	58.63	68.58	500,000	584,901	84,901	16.98%	1.00%
ARCELORMITTAL	25,157	19.88	16.91	500,000	425,532	–74,468	–14.89%	1.00%
BASF	20,559	24.32	26.05	500,000	535,568	35,568	7.11%	1.00%
BAYER	11,468	43.60	49.47	500,000	567,369	67,369	13.47%	1.00%
Industrials								
ALSTOM	12,028	41.57	44.30	500,000	532,839	32,839	6.57%	1.00%
SAINT GOBAIN	16,562	30.19	28.58	500,000	473,390	–26,610	–5.32%	1.00%
SCHNEIDER ELECTRIC	9,349	53.48	52.39	500,000	489,844	–10,156	–2.03%	1.00%
SIEMENS	21,295	46.96	52.10	1,000,000	1,109,447	109,447	10.94%	2.00%
VINCI	17,241	29.00	28.64	500,000	493,720	–6,280	–1.26%	1.00%
Consumer Goods								
DAIMLER	20,851	23.98	22.66	500,000	472,464	–27,536	–5.51%	1.00%
GRP DANONE	13,041	38.34	31.42	500,000	409,743	–90,257	–18.05%	1.00%
L'OREAL	9,465	52.83	54.03	500,000	511,429	11,429	2.29%	1.00%
LVMH MOET HENNESSY	10,209	48.98	43.53	500,000	444,460	–55,540	–11.11%	1.00%
PHILIPS ELECTRONICS	33,190	15.07	16.26	500,000	539,762	39,762	7.95%	1.00%
RENAULT	30,441	16.43	17.36	500,000	528,538	28,538	5.71%	1.00%
UNILEVER NV	30,798	16.24	16.30	500,000	502,041	2,041	0.41%	1.00%
VOLKSWAGEN	1,963	254.74	231.63	500,000	454,640	–45,360	–9.07%	1.00%
Health Care								
SANOFI-AVENTIS	106,758	46.84	42.67	5,000,000	4,554,995	–445,005	–8.90%	10.00%
Consumer Services								
CARREFOUR SUPERMARCHE	18,090	27.64	24.57	500,000	444,535	–55,465	–11.09%	1.00%
VIVENDI	24,988	20.01	20.80	500,000	519,723	19,723	3.94%	1.00%
Telecommunications								
DEUTSCHE TELEKOM	51,335	9.74	8.68	500,000	445,470	–54,530	–10.91%	1.00%
FRANCE TELECOM	27,137	18.43	20.71	500,000	562,018	62,018	12.40%	1.00%

Figure 8.4: Example valuation report.

Lab Exercise 8.1: Portfolio Valuation

1. Open the file **PortfolioValuation.sql** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the ORACLE PL/SQL code illustrating the performance calculation shown above.
2. Using Oracle run the following examples at your Oracle SQL Plus command line using the example database for this book you created from the scripts downloaded from the companion web site:

```
SQL> @PortfolioValuation.sql

...

SQL> InitialisePtfValuation()

...

SQL> RunPtfValuation()

...

SQL> PtfValuationReport()
```

8.2.2 Simple Performance

Simple performance or the return for a single period is easy to calculate; you already saw the formula illustrated in the previous section. Since it is such a basic and fundamental calculation we have included a more detailed version of it in Formula 8-1. The key is that simple performance should be calculated for a period that does not include portfolio in- or outflows. Whenever there are any in- or outflows you should calculate the simple performance between each cash flow and ideally use the chain linking formula for Time-Weighted Performance shown later to arrive at a result. Or, if that is not possible, use either a money-weighted or (much better) a modified Dietz approach, also shown later. You can also see the Formula 8-1 implemented in the SQL from Lab Exercise 8.1.

Formula 8-1: Simple portfolio return

$$Y_{Simple} = \frac{V_{t+1} - V_t + \sum_{i=1}^{i=n}[I_i]}{V_t}$$

$$Y_{Simple} = \left(V_{Period\ End} - V_{Period\ Start}\right) / V_{Period\ Start}$$

$$Y_{Simple.in\ Percent} = Y_{Simple} \times 100$$

where:

Y_{Simple}: Money weighted return for the portfolio in decimal format

$Y_{Simple.\ in\ Percent}$: Money weighted return in percent

$V_{Period\ Start}$: Market value of the portfolio at the beginning of the period

$V_{Period\ End}$: Market value of the portfolio at the end of the period

$\sum_{i=1}^{i=n}[I_i]$: Net income and expenses for the portfolio

8.2.3 Money-Weighted Performance

Money-Weighted Performance is a commonly used analysis tool in finance. The problem with Money-Weighted Performance is that it does not give very stable performance results.

Formula 8-2: Money-weighted portfolio return

$$Y_{ApprocMW} = \frac{V_{t+1} - V_t + \sum_{i=1}^{i=n}[I_i] - \sum_{j=1}^{j=m}[F_j]}{V_t + \sum_{j=1}^{j=m}[F_j]}$$

$$Y_{MWbyIRR} = IRR\left(\left[V_t + \sum_{j=1}^{j=m}[F_j]\right], \left[\sum_{i=1}^{i=n}[I_i], V_{t+1}\right]\right)$$

where:

$Y_{ApprocMW}$: Approximate money-weighted return for the portfolio

$Y_{MWbyIRR}$: Money-weighted portfolio return using the IRR function

V_t: Market value of the portfolio at the beginning of the period

V_{t+1}: Market value of the portfolio at the end of the period

$\sum_{i=1}^{i=n}[I_i]$: Net income and expenses for the portfolio

$\sum_{i=1}^{i=n}[F_i]$: Net capital in or outflows from the portfolio

In Formula 8-2 we have set out the details for calculating money-weighted performance by approximation for a given period. Work through Lab Exercise 8.2 to see the difference in the money-weighted performance in the example portfolios A and B. You may also want to compare the results to those from Lab Exercise 8.3 on time-weighted performance.

Lab Exercise 8.2: Money-Weighted Portfolio Return

1. Open the file **MoneyWeightedPerformance.sql** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the ORACLE PL/SQL code illustrating the performance calculation shown above.
2. Using Oracle run the following examples at your Oracle SQL Plus command line using the example database for this book you created from the scripts downloaded from the companion web site:

```
SQL> @MoneyWeightedPerformance.sql

…

SQL> InitialiseMWPerformance()

…

SQL> RunMWPerformance ()

…

SQL> MWPerformanceReport()
```

8.2.4 Time-Weighted Performance

If you worked through Lab Exercise 8.2 you saw that a portfolio's money-weighted performance can be significantly influenced by the timing of capital in- or outflows to the portfolio. This is not very desirable since it over- or underrates the true results of a portfolio manager if there are any such in- or outflows. This easily could mislead an investor taking wrong decisions based on such distorted performance data.

Fortunately, this drawback of money-weighted performance is overcome by time-weighted performance. In Formula 8-3 you can see how to link together the results from individual simple performance measures calculated using Formula 8-1 or the modified Dietz Method shown later for each performance period. The length of a performance period should ideally be the period between any cash in- or outflows, or better still, daily.

Formula 8-3: Cumulative portfolio returns

$$Y_{\text{Period 1 to n}} = \left[(1+Y_1)\times(1+Y_2)\times\ldots\times(1+Y_n)\right]-1$$

or:

$$Y_{\text{Period 1 to n}} = \left[\prod_{t=1}^{t=n}(1+Y_t)\right]-1$$

or:

$$Y_{\text{Period 1 to n}} = \left[(1+Y_{\text{Period 1 to n}-1})\times(1+Y_1)\right]-1$$

where:

$Y_{\text{Period 1 to n}}$: Chain-linked cumulative return for period **1** to **n**

$Y_{\text{Period 1 to n-1}}$: Chain-linked cumulative return for period **1** to **n − 1**

Y_1, Y_2, Y_n, Y_t: Period return for Period 1, Period 2, Period **n**, and Period **t**

One problem with the chain-linked time-weighted returns is that the overall period may vary from portfolio to portfolio for many reasons, such as that they started at different times in history. To solve this we can transform the return into a standardised period for easier comparison. In Formula 8-4 we show the most popular transformation to an annualised return rate. You can use this formula to obtain an annualised return rate for different portfolios and then compare their relative attractiveness using annualised portfolio returns for each.

Formula 8-4: Annualised portfolio return

$$Y_{\text{Annualized}} = \left[(1+Y_T)^{1/\text{Years}(T)}\right]-1$$

where:

$Y_{\text{Annualized}}$: Annualised return for period T

Y_T: Cumulative return for period T

Years(T): Number of years (including fractions) in Period T

We have pointed out that it is always preferable to measure and record the performance of a portfolio between any in- and outflows. This even should include the receipt of income such as coupon payments or expenses deducted. In some cases, however, including such intermediate cash flows may turn out to be impractical. In this case you can use the Modified Dietz approach using Formula 8-5 instead of the simple performance calculation from Formula 8-1.

Formula 8-5: Modified Dietz return

$$Y_{\text{Modified Dietz}} = \left(V_{\text{End}} - V_{\text{Start}} - F_{\text{Inflows}} - F_{\text{Outflows}}\right) / \left(V_{\text{Start}} + F_{\text{Net Time Weighted}}\right)$$

$$F_{\text{Net Time Weighted}} = \sum_{t=1}^{t=n} \left[\left(\tau_t / \tau_{\text{Period}}\right) \times F_t\right]$$

where:

$Y_{\text{Modified Dietz}}$: Modified Dietz time return for the portfolio in decimal format

V_{Start}: Market value of the portfolio at the beginning of the period

V_{End}: Market value of the portfolio at the end of the period

F_{Inflows}: Sum of new funds added to the portfolio during the period

F_{Outflows}: Sum of funds withdrawn from the portfolio during the period

$F_{\text{Net Time Weighted}}$: Net sum of time-weighted fund flows in and out of the portfolio

τ_t: Number of days from start of the period to cash flow **t**

τ_{Period}: Total number of days in period for calculating the return

F_t: Cash flow **t** for **t** in 1 ... n

n: Total number of all fund flows in and out of the portfolio

If you examine Formula 8-5 more closely you will see that the Modified Dietz approach is a compromise between time- and money-weighted returns. In fact, it simply assumes that the cash flows on average take place in the middle of the period, which for many small randomly distributed in- and outflows over the period is not an unreasonable assumption.

To see both time-weighted and Modified Dietz in action you should work through Lab Exercise 8.3, which is an example of how to implement both and also will help you appreciate the nature of them. If you can, you should compare your results from Exercise 8.3 with those of Exercise 8.2 as this will help you better understand the behaviour of all three, time-weighted, money-weighted, and the compromise Modified Dietz approach.

Lab Exercise 8.3: Time-Weighted Portfolio Return

1. Open the file **TimeWeightedPerformance.sql** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the ORACLE PL/SQL code illustrating the performance calculation shown above.
2. Using Oracle run the following examples at your Oracle SQL Plus command line using the example database for this book you created from the scripts downloaded from the companion web site:

```
SQL> @TimeWeightedPerformance.sql

…

SQL> InitialiseTWPerformance()

…

SQL> RunTWPerformance ()

…

SQL> TWPerformanceReport()
```

8.2.5 Risk Adjusted Performance

An important reason for carrying out performance measurement is to compare different portfolios in terms of how well they are managed. By using Time-Weighted Performance measures, and in particular standardised rates such as annualised performance, it is possible to compare how well portfolios have fared in the past. What is left out by these measures are the risks that were taken. It is quite possible that the portfolio with seemingly stellar performance was not well managed by the portfolio manager, but simply achieved the results by taking higher than average risks and being lucky in that during the period concerned. The risk only showed itself in the form of positive upswings, cloaking the true risk taken to some degree and accidentally sparing the portfolio manager from painful draw-downs and substantial negative returns.

If you want to compare portfolios that are different in risk, you will need is risk sensitive measures of return. In Figure 8.5 we show a useful classification of the most common risk adjusted portfolio performance measures. The diagram has been adapted from a paper by Scholz and Wilkens (2005), which first proposed this classification. It is useful to walk

Classification of Portfolio Performance Measures *adapted from Scholz & Wilkens [2005]*	Performance Ratios based on Total Risk	Performance Ratios based on Market Risk
Ratio of excess portfolio return to portfolio risk	Sharpe Ratio	Treynor Ratio
Differential return between portfolio and risk adjusted benchmark	Total Risk Alpha	Jensen Alpha
Return of the risk adjusted portfolio	$RAP_{Total\ Risk}$	$RAP_{Market\ Risk}$
Differential return of the risk adjusted portfolio	$DR_{Total\ Risk}$	$DR_{Market\ Risk}$

Figure 8.5: Classification of risk-adjusted portfolio performance measures.

through the different measures in turn, making it easier to choose the right measure for a given task. We will now briefly cover each measure in turn.

8.2.5.1 Sharpe Ratio

The Sharpe Ratio is a measure of the risk adjusted return, which we already looked at in Chapter 6. We reproduce its definition again here in Formula 8-6. The Sharpe Ratio is measured based on total portfolio risk and shows the excess portfolio return in proportion to the portfolio's total risk. For a given return, the greater the risk of a portfolio the smaller the ratio. It can therefore be used to rank different portfolios by their combined risk return profile.

Formula 8-6: Sharpe ratio

$$S = \frac{R_P - R_{RF}}{\sigma_P}$$

where:

R_P: Annualised portfolio return

σ_P: Annualised standard deviation of the portfolio return

R_{RF}: Risk-free interest rate

Formula 8-7: Total risk Alpha

$$\alpha_{Total} = R_P - R_{BxP}$$

using:

$$R_{BxP} = R_{RF} + \frac{\sigma_P \times [R_B - R_{RF}]}{\sigma_B}$$

where:

α_{Total}: Total risk alpha

R_P: Portfolio return

R_B: Market return

R_{RF}: Risk-free rate of return

R_{BxP}: Return of benchmark adjusted by the risk of the portfolio

σ_P: Annualised standard deviation of the portfolio return

σ_B: Annualised standard deviation of the portfolio return

Formula 8-7 sets out the calculation for total risk alpha. This is a measure of the differential return between a given portfolio and an equivalent benchmark portfolio adjusted to match the total risk of the portfolio under analysis.

An alternative to total risk alpha and better suited are the two measures of Risk Adjusted Performance (RAP) for total risk and Differential Return (DR) for total risk. The calculations for both are set out in Formulas 8-8 and 8-9, respectively.

Risk Adjusted Performance (RAP) gives you the performance of the portfolio adjusted for the total risk of the portfolio. This makes the Risk Adjusted Performance a measure by which you can compare the performance between portfolios with very different risk profiles.

Differential Return on the other hand gives you the difference in risk adjusted returns between the portfolio under analysis and an appropriate market or benchmark portfolio. It is thus derived from Risk Adjusted Performance itself.

Formula 8-8: RAP risk adjusted performance

$$\Phi_{Total} = \left[(\sigma_B \cdot \; S)R_{RF}\right]$$

using:

$$S = \text{Sharpe Ratio} = \frac{R_P - R_{RF}}{\sigma_{P,R}}$$

where:

Φ_{Total}: Risk-adjusted performance based on total risk

σ_B: Estimated standard deviation of the returns of the portfolio

S: Sharpe ratio

R_{RF}: Risk-free rate of return

R_P: Annualised portfolio return

$\sigma_{P,R}$: Annualised standard deviation of the portfolio return

Formula 8-9: Differential return based on total risk

$$\Delta_{Total} = R_{RF} + [(E(R_B) - R_{RF}) \times \sigma_P]/\sigma_B$$

where:

δ_{Market}: Differential return (see Elton & Gruber, 1995, pp. 641)

R_{RF}: Risk-free rate of return

$E(R_B)$: Expected return for the benchmark

σ_B: Estimated standard deviation of the returns of the portfolio

σ_B: Estimated standard deviation of the returns of the benchmark

If you now look at the right-hand column of models in Figure 8.5 you will see that the Treynor Ratio, which we first introduced in Chapter 6, is the equivalent of the Sharpe Ratio but using market risk instead of total risk in its calculation. We have reproduced the Treynor Ratio here again for convenience in Formula 8-10.

Formula 8-10: Treynor ratio

$$T = \frac{R_P - R_{RF}}{\beta_P}$$

using:

$$\beta_{P,B} = cov(R_P, R_B) / \sigma^2 B$$

where:

R_P = Annualised portfolio return

R_{RF} = Risk-free interest rate

β_P = Beta of portfolio **P** with respect to the benchmark **B**

In Formula 8-11 you see the details for calculating Jensen Alpha, which is the market risk equivalent of the Total Risk Alpha. As such it gives us the differential return between the market and the portfolio under analysis adjusted for market risk.

Formula 8-11: Jensen Alpha

$$\alpha_{Jensen} = R_P - E(R) \times \beta_P$$

using:

$$E(R_P) = R_{RF} + \beta_P \times [E(R_B) - R_{RF}]$$

$$\beta_P = [COV(R_P, R_B)] / \sigma^2_B$$

where:

α_{Jensen}**:** Jensen Alpha

R_P**:** Portfolio return as measured

$E(R_P)$**:** Expected return for the portfolio

β_P**:** Beta of portfolio **P** with respect to the benchmark **B**

$E(R_B)$**:** Expected return for the benchmark

R_{RF}**:** Risk-free rate of return

$COV(R_P, R_B)$**:** Estimated covariance between benchmark and portfolio returns

σ^2_B**:** Estimated std deviation of the returns of the benchmark

Again, as in the case for total risk, you might find it preferable to use the Risk Adjusted Performance and Differential Return instead of Jensen's Alpha for portfolio comparisons. Market Risk Adjusted Performance gives you the return for a portfolio that is adjusted for market risk using its Beta relative to a suitably chosen market index or benchmark.

Differential Return based on market risk is again based on Market Risk Adjusted Performance and gives you the difference between the risk adjusted returns between the portfolio and a market index or benchmark.

Formula 8-12: MRAP risk adjusted performance

$$\varphi_{Market} = \left[R_P \times \left(1/\beta_P - 1\right) \times \left(R_P - R_{RF}\right)\right]$$

where:

φ_{Market}: Risk-adjusted performance based on market risk

σ_B: Estimated standard deviation of the returns of the portfolio

R_{RF}: Risk-free rate of return

R_P: Annualised portfolio return

β_P: Beta of portfolio **P** with respect to the benchmark **B**

Formula 8-13: Differential return based on market risk

$$\Delta_{Market} = \frac{\alpha_{Jensen}}{\beta_P}$$

where:

Δ_{Market}: Differential return based on market risk

α_{Jensen}: Jensen Alpha

β_P: Beta of portfolio **P** with respect to the benchmark **B**

Before we now jump to performance attribution it is worthwhile to work through Lab Exercise 8.4. Analysing the code and the calculation results will give you a better understanding of the subtleties when using one measure or the other.

Lab Exercise 8.4: Risk-Adjusted Portfolio Performance

1. Open the file **RiskAdjustedPerformance.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the performance calculation shown above.
2. Using the R application load the file **RiskAdjustedPerformance.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseRAPerformance()

...

R> RunRAPerformance ()

...

R> RAPerformanceReport()
```

8.3 Performance Attribution

In the preceding sections we introduced the tools needed to measure and compare different portfolios both by their return over time and by taking into account their individual risk profiles. As a portfolio manager, and often as an investor too, it is very useful to take a closer look at the insides of a portfolio. That is, we often need to understand better where a portfolio's results are coming from, and by the same token, which decision led to a given set of results relative to a benchmark. This is the domain of Performance Attribution.

Before we can begin with the topic of performance attribution proper, we need to backtrack to the topic of benchmarks as Performance Attribution heavily relies on them. We already have used benchmarks in the previous section on risk adjusted performance, but it is now time to explore the construction of benchmarks in move detail before continuing to use them.

8.3.1 Benchmarks

A benchmark-or-index is very similar to a portfolio. In fact, indices are benchmarks of financial instruments frequently designed to stand in as an equivalent to a portfolio. This is particularly true of performance benchmarks and indices. They have been purposely designed so that the portfolio performance can be measured against them. While in a portfolio, the positions and hence the relative weight of instruments are based on the portfolio manager's discretion, the relative weight of instruments in an index follows a defined (and published) rule. The weight is most commonly determined by the instrument market capitalisation or free float. The market capitalisation is the total number of units of an instrument times the price of the instrument. The free float is the number of units of an instrument that are freely tradable times the price of the instrument. The instruments used in the calculation of an index are called index constituents.

In Formula 8-14 we set out the details for calculating the value of an equity index for a given day from underlying data like market prices. The weight for indices or benchmarks (M in the formula) usually is reset only at certain intervals. For equities, this can be once a month, once a quarter, or once or twice in a year.

Formula 8.14: Market capitalisation weighted equity index

$$P_0^{Equity} = B$$

$$P_t^{Equity} = \frac{B \times L_t \times \sum_{i=1}^{i=n}\left[P_{i,t} \times M_{i,t-1} \times a_{i,t}\right]}{\sum_{i=1}^{i=n}\left[P_{i,t-1} \times M_{i,t-1}\right]}$$

where:

P_0^{Equity}**:** Value of the equity index at inception

P_t^{Equity}**:** Value of the equity index at any point after inception

$P_{i,t}$**:** Price of equity **i** at time **t**

$P_{i,t-1}$**:** Price of equity **i** at time **t – 1**

$a_{i,t}$**:** Adjustment factor of equity **i** at time **t**

$M_{i,t-1}$**:** Market capitalisation or Free Float of equity **i** at time **t – 1**

L_t**:** Chain-link factor for the index at time **t**

B: Index base at inception

n: Number of index constituents

One of the inputs in Formula 8-14 is the chain-link factor for the index. This factor ensures that the index values link together smoothly from one day to the next when the index weights are readjusted in between. That is, with different weights in the index and possibly different constituents the value of the index would jump on days following a change in constituents or weight. The chain-link factor compensates for this and ensures that if the new composition and weighting was applied for the last day, the index was calculated with the old composition and weights, the index value would be exactly the same as if calculated with the old composition and weights.

This makes the index values from before the change directly comparable to those after the change. The calculation of the chain-link factor is shown in Formula 8-15.

Formula 8-15: Chain link factor for an equity index

$$\mathbf{L}_{t+1} \times \sum_{i=1}^{i=n} \left[\mathbf{P}_{i,t} \times \mathbf{M}_{i,t+1} \times \mathbf{a}_{i,t} \right] = \mathbf{L}_t \times \sum_{i=1}^{i=n} \left[\mathbf{P}_{i,t} \times \mathbf{M}_{i,t} \times \mathbf{a}_{i,t} \right]$$

and thus:

$$\mathbf{L}_{t+1} = \frac{\mathbf{L}_t \times \sum_{i=1}^{i=n} \left[\mathbf{P}_{i,t} \times \mathbf{M}_{i,t} \times \mathbf{a}_{i,t} \right]}{\sum_{i=1}^{i=n} \left[\mathbf{P}_{i,t} \times \mathbf{M}_{i,t+1} \times \mathbf{a}_{i,t} \right]}$$

where:

$\mathbf{L}_t$: Chain-link factor for the index at time **t**

$\mathbf{L}_{t+1}$: Chain-link factor for the index at time **t + 1**

$\mathbf{P}_{i,t}$: Price of equity **i** at time **t**

$\mathbf{P}_{i,t-1}$: Price of equity **i** at time **t – 1**

$\mathbf{a}_{i,t}$: Adjustment factor of equity **i** at time **t**

$\mathbf{M}_{i,t-1}$: Market capitalisation or Free Float of equity **i** at time **t – 1**

$\mathbf{L}_t$: Chain-link factor for the index at time **t**

n: Number of index constituents

Another input in Formula 8-14 that requires explanation is the Adjustment Factor for equities in the index. This is needed if a dividend is being paid by the equity share. The Adjustment Factor adjusts the index for the price drop that an equity experiences on the day after the dividend was last included in the price or, in other words, the day after it goes ex-dividend. The calculation of the Adjustment Factor is shown in Formula 8-16. The Adjustment Factor normally is reset to zero for each instrument each time the index constituents and weights are subsequently changed.

Formula 8-16: Adjustment factor for an equity in an equity index

$$\mathbf{a}_{i,t} = \frac{\mathbf{P}_{i,t-1}}{\mathbf{P}_{i,t-1} - \mathbf{D}_{i,t}}$$

where:

$\mathbf{a}_{i,t}$: Adjustment factor of equity **i** at time **t**

$\mathbf{P}_{i,t-1}$: Price of equity **i** at time **t – 1**

$\mathbf{D}_{i,t}$: Dividend for equity **i** at time **t**

To bring benchmark calculation to life we have put together a lab exercise that utilises this formula and some small real-life index composition data sampled from the STOXX indices.

Once you have worked through the lab exercise we would encourage you to go to the STOXX web site (www.stoxx.com), sign up to a free login and take advantage of the free real data downloads available on the site. Use them as further practice data that you can download into your lab database.

Lab Exercise 8.5: Equity Benchmark Calculation

1. Open the file **EquityBenchmarkCalc.sql** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the ORACLE PL/SQL code illustrating the performance calculation shown above.
2. Using Oracle run the following examples at your Oracle SQL Plus command line using the example database for this book you created from the scripts downloaded from the companion web site:

```
SQL> @EquityBenchmarkCalc.sql

...

SQL> InitialiseEquBenchmark()

...

SQL> RunEquBenchmark()

...

SQL> EquBenchmarkReport()
```

8.3.2 Absolute Attribution

The first step in attribution is to determine how much a particular instrument position in a portfolio or benchmark has contributed to the overall return of the portfolio or benchmark. This can be very easily calculated from the data we derived during the valuation. As set out in Formula 8-17, the contribution in absolute terms of the overall portfolio or benchmark return is simply the return of the position over the period times its weight.

Formula 8-17: Absolute contribution of an individual position in base currency

$$C_{I,P} = W_{I,P} \times R_{I,U}$$

$$C_{I,B} = W_{I,B} \times R_{I,U}$$

where:

$C_{I,P}$, $C_{I,B}$: Contribution of an individual instrument (I) position to the return of the portfolio (P) or benchmark (B) as a whole or a segment in single level or hierarchical segmental analysis

$W_{I,P}$, $W_{I,B}$: Weight of a given individual instrument (I) position in a portfolio (P) or benchmark (B)

$R_{I,U}$: Return of an individual instrument position (I) in portfolio (P) or benchmark (B) base currency

In Figure 8.6 you can see an example performance attribution report that details the contribution of each position to the portfolio as a whole. This Absolute Attribution is shown in the section attribution by Security in the column title of **Contrib**.

8.3.3 Relative Attribution

The example also shows the relative attribution of the performance at both an instrument and an industry level, in the column labelled **Attribution**. This **Relative Attribution** is carried out over two levels. The effect of security selection on the performance is calculated and shown at the instrument position level. The effect of asset allocation on an industry basis in this case is calculated and shown on an industry level. Although in the example this has been done over two levels, this could be carried out for any number of levels. In Formula 8-18 we set out the details for calculating the component attribution effect shown in Figure 8.6.

8.3.3.1 Brinson and Fachler Component Performance Attribution

The calculation in Formula 8-18 is based on the method proposed by Brinson and Fachler (BF) in the seminal paper in 1985 that is the basis of many modern attribution systems.

bancstreet
capital partners

Portfolio Performance Report
Client XYZ - Sample Portfolio

Position Performance Attribution

Analysis by Factor

Client: XYZ General Insurance Plc
Period: 01.01.2010 to 31.01.2010
Ref. Currency: GBP

Portfolio ID: XYZ-GEN-EQU-BAL
Client-Manager: Sally Sample-Smith
Benchmark: XYZ-GEN-ALM-BMK

Attribution by Industry

	Weight	Bench	U/O W	Contr	% Contr	Return	Attr	Value
Oil & Gas	26.00%	11.94%	14.06%	–0.25%	21.48%	–0.25%	0.13%	12,846,912
Basic Materials	4.00%	7.37%	–3.37%	–0.13%	14.23%	–0.13%	–0.03%	396,304
Industrials	6.00%	7.16%	–1.16%	0.05%	0.37%	0.05%	0.15%	339,005
Consumer Goods	8.00%	4.17%	3.83%	–0.25%	1.70%	–0.25%	0.56%	340,683
Health Care	10.00%	3.46%	6.54%	–1.48%	26.95%	–1.48%	0.26%	236,244
Consumer Services	2.00%	11.43%	–9.43%	0.10%	–54.42%	0.10%	–0.03%	49,910
Telecommunications	13.00%	11.43%	1.57%	1.14%	–27.60%	1.14%	0.08%	311,523
Utilities	9.00%	12.36%	–3.36%	–0.21%	6.51%	–0.21%	–0.10%	87,159
Financials	20.00%	25.00%	–5.00%	–0.22%	2.76%	–0.22%	–0.40%	91,866
Technology	2.00%	5.35%	–3.35%	0.08%	–4.05%	0.08%	–0.07%	9,796
ALL INDUSTRIES	100.00%	100.00%	0.00%	–1.18%	–12.06%	–1.18%	0.57%	14,709,402

Attribution by Security

Name	Position	Price	Value	Weight	U/O W	Return	Contrib	Attribution
			49,411,201	100%	0.00%	–1.18%	100.00%	–0.01%
TOTAL	244,349	40.55	9,907,589	20.00%	12.60%	-0.92%	15.69%	0.07%
TELEFONICA	351,124	16.09	5,650,091	10.00%	4.97%	13.00%	–110.41%	0.06%
SANOFI-AVENTIS	106,758	39.89	4,258,539	10.00%	5.83%	–14.83%	125.93%	0.00%
BCO SANTANDER	431,034	5.48	2,362,438	5.00%	1.01%	–5.50%	23.36%	0.00%
E.ON	104,734	23.83	2,495,512	5.00%	1.21%	–0.18%	0.76%	0.01%
ENI	146,199	16.39	2,396,327	5.00%	1.41%	–4.15%	17.61%	0.01%
SIEMENS	21,295	45.48	968,435	2.00%	–1.30%	–3.16%	5.36%	–0.01%
NOKIA	51,282	8.96	459,332	1.00%	–2.19%	–8.13%	6.91%	–0.01%
GDF SUEZ	17,781	27.55	489,795	1.00%	–2.13%	–2.04%	1.73%	–0.01%
FRANCE TELECOM	27,137	18.85	511,542	1.00%	–2.01%	2.31%	–1.96%	–0.01%
BAYER	11,468	45.58	522,699	1.00%	–1.87%	4.54%	–3.86%	–0.01%
ALLIANZ	7,520	70.39	529,354	1.00%	–1.59%	5.87%	–4.99%	–0.01%
DEUTSCHE TELEKOM	51,335	8.44	433,436	1.00%	–1.50%	–13.31%	11.31%	0.00%
VOLKSWAGEN	1,963	257.11	504,657	1.00%	–1.40%	0.93%	–0.79%	–0.01%
UNILEVER NV	30,798	17.26	531,482	1.00%	–1.17%	6.30%	–5.35%	–0.01%
BCO BILBAO VIZCAYA ARGENTARIA	70,621	6.63	468,264	1.00%	–1.17%	–6.35%	5.39%	–0.01%
SAP	17,361	33.52	581,994	1.00%	–1.17%	16.40%	–13.93%	–0.01%
RWE	8,521	49.65	423,058	1.00%	–1.08%	–15.39%	13.07%	0.00%

tools.bancstreet.com Pricing, Risk and Performance Measurement in Practice

Figure 8.6: Example position performance attribution.

Formula 8.18: Component attribution effect (Brinson and Fachler, 1985)

$$A_S = \left[\left(W_{S,P} - W_{S,B}\right) \times \left(R_{S,B} - R_B\right)\right]$$

using:

$$W_{S,P} = \sum_{k=1}^{n} W_k$$

$$W_{S,B} = \sum_{g=1}^{n} W_g$$

$$R_{S,B} = \sum_{g=1}^{n} W_g \times R_g = \sum_{g=1}^{n} C_g$$

$$R_B = \sum_{j=1}^{n} W_j \times R_j = \sum_{g=1}^{n} C_j$$

where:

A_S: Allocation effect for sector **S** in portfolio or parent segment **P**

$W_{S,P}$: Weight of segment **S** in relation to portfolio or parent segment **P** as a whole in percent

$W_{S,B}$: Weight of segment **S** in relation to benchmark or parent segment **B** as a whole in percent

$R_{S,B}$: Absolute contribution of segment **S** in the benchmark or parent segment **B** to the benchmark or parent segment as a whole

R_B: Return of the benchmark or parent segment as a whole

W_k, W_g, W_j: Weight of instrument **k** or **g** or **j**, respectively, selected from instrument **1** to **n** in a segment, portfolio, or benchmark in relation to the parent segment, portfolio, or benchmark as a whole

R_k, R_g, R_j: Return of instrument **k** or **g** or **j**, respectively, selected from instrument **1** to **n** in a segment, portfolio, or benchmark in relation to the parent segment, portfolio, or benchmark as a whole

n: Number of instruments in a segment, portfolio, or benchmark, respectively, when analysing a segment, portfolio, or benchmark

The best way to explore the workings of attribution is again with real data. Use Lab Exercise 8.6 to apply the calculation to an example portfolio. Once you have worked through the lab, ideally go and change the portfolio example data so that you can see the changes with different scenarios for portfolio composition and returns.

Lab Exercise 8.6: Brinson & Fachler Component Performance Attribution

1. Open the file **BFComponentAttrib.sql** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the ORACLE PL/SQL code illustrating the performance calculation shown above.
2. Using Oracle run the following examples at your Oracle SQL Plus command line using the example database for this book you created from the scripts downloaded from the companion web site:

```
SQL> @BFComponentAttrib.sql

…

SQL> InitialiseBFComponentAttrib ()

…

SQL> RunBFComponentAttrib ()

…

SQL> BFComponentAttribReport()
```

8.3.3.2 Brinson, Hood, and Beebower Portfolio Performance Attribution

The classical approach to performance attribution at a portfolio level was proposed by Brinson, Hood, and Beebower (BHB) in their 1986 paper.

Figure 8.7 shows an example report using BHB portfolio attribution. The formula set for the BHB attribution method is a little more involved. The first step is to calculate the Allocation Effect shown in the column titled "Ass. AU.". This gives you the effect that asset allocation decisions have on differential performance between the portfolio and the benchmark. The details for calculating the BHB Allocation effect are set up in Formula 8-19.

bancstreet
capital partners

Portfolio Performance Report
Client XYZ - Sample Portfolio

Performance Attribution by Industry

Analysis by Factor

Client: XYZ General Insurance Plc
Period: 01/01/2010 to 31/01/2010
Ref. Currency: GBP

Portfolio ID: XYZ-GEN-EQU-BAL
Client-Manager: Sally Sample-Smith
Benchmark: XYZ-GEN-ALM-BMK

Total Portfolio Attribution Analysis		
Asset Allocation	0.21%	–25.04%
Security Selection	–0.50%	58.90%
Interaction Effect	–0.56%	66.14%
Total Effect	–0.85%	100.00%

Portfolio by Industry

	Initial Value	Final Value	Gain/Loss	Return	Weight	Bench	Rel Wgt
Oil & Gas	1,646,115 €	1,544,269 €	–101,846 €	–6.19%	3.29%	11.94%	–8.64%
Basic Materials	4,130,000 €	4,018,658 €	–111,342 €	–2.70%	8.26%	7.37%	0.89%
Industrials	5,071,795 €	5,043,965 €	–27,830 €	–0.55%	10.14%	7.16%	2.98%
Consumer Goods	8,731,740 €	8,178,699 €	–553,041 €	–6.33%	17.46%	11.75%	5.72%
Health Care	500,000 €	540,675 €	40,675 €	8.14%	1.00%	4.17%	–3.17%
Consumer Services	1,500,000 €	1,691,895 €	191,895 €	12.79%	3.00%	3.46%	–0.46%
Telecommunications	4,449,445 €	4,105,402 €	–344,043 €	–7.73%	8.90%	11.43%	–2.53%
Utilities	5,500,000 €	5,670,110 €	170,110 €	3.09%	11.00%	12.36%	–1.36%
Financials	16,469,730 €	16,056,348 €	–413,382 €	–2.51%	32.94%	25.00%	7.94%
Technology	2,000,000 €	1,940,569 €	–59,431 €	–2.97%	4.00%	5.35%	–1.35%
ALL INDUSTRIES	49,998,825 €	48,790,591 €	–1,208,234 €	–2.42%	100.00%	100.00%	0.00%

Attribution by Industry

	Contrib.	Ass. All.	Sec. Sel.	Residual	Total	Weight	Rel Wgt
Oil & Gas	8.43%	0.3041%	0.3392%	–0.2457%	0.3976%	3.29%	–8.64%
Basic Materials	9.22%	–0.0098%	0.0273%	0.0033%	0.0207%	8.26%	0.89%
Industrials	2.30%	–0.0297%	0.0582%	0.0242%	0.0527%	10.14%	2.98%
Consumer Goods	45.77%	0.0578%	–0.5498%	–0.2674%	–0.7594%	17.46%	5.72%
Health Care	–3.37%	–0.0356%	–0.0356%	0.0271%	–0.0442%	1.00%	–3.17%
Consumer Services	–15.88%	–0.0063%	–0.0029%	0.0004%	–0.0088%	3.00%	–0.46%
Telecommunications	28.47%	0.0527%	–0.0232%	0.0052%	0.0346%	8.90%	–2.53%
Utilities	–14.08%	–0.0161%	–0.0063%	0.0007%	–0.0217%	11.00%	–1.36%
Financials	34.21%	–0.1151%	–0.3231%	–0.1026%	–0.5408%	32.94%	7.94%
Technology	4.92%	0.0099%	0.0180%	–0.0045%	0.0233%	4.00%	–1.35%
ALL INDUSTRIES	100.00%	0.2118%	–0.4983%	–0.5595%	–0.8459%	100.00%	0.00%

tools.bancstreet.com   Pricing, Risk and Performance Measurement in Practice

Figure 8.7: Example performance attribution report.

Formula 8-19: Allocation effect

$$A_S = \left[\left(W_{S,P} - W_{S,B} \right) \times \left(R_{S,B} - R_B \right) \right]$$

using:

$$W_{S,P} = \sum_{k=1}^{n} W_k$$

$$W_{S,B} = \sum_{g=1}^{n} W_g$$

$$R_{S,B} = \sum_{g=1}^{n} W_g \times R_g = \sum_{g=1}^{n} C_g$$

$$R_B = \sum_{j=1}^{n} W_j \times R_j = \sum_{g=1}^{n} C_j$$

where:

A_S: Allocation effect for sector **S** in portfolio or parent segment **P**

$W_{S,P}$: Weight of segment **S** in relation to portfolio or parent segment **P** as a whole in percent

$W_{S,B}$: Weight of segment **S** in relation to benchmark or parent segment **B** as a whole in percent

$R_{S,B}$: Absolute contribution of segment **S** in the benchmark or parent segment B to the benchmark or parent segment as a whole

$R_{B,}$: Return of the benchmark or parent segment as a whole

W_k,W_g,W_j: Weight of instrument **k** or **g** or **j**, respectively, selected from instrument **1** to **n** in a segment, portfolio, or benchmark in relation to the parent segment, portfolio, or benchmark as a whole

R_k,R_g,R_j: Return of instrument **k** or **g** or **j**, respectively, selected from instrument **1** to **n** in a segment, portfolio, or benchmark in relation to the parent segment, portfolio, or benchmark as a whole

n: Number of instruments in a segment, portfolio, or benchmark, respectively, when analysing a segment, portfolio, or benchmark

The second step in the BHB Method is to calculate the effect of security selection. This selection effect is shown in column “See. Sel.” of Figure 8.7. Its calculation is set out in Formula 8-20.

Formula 8-20: Selection effect

$$S_S = \left[\left(R_{S,P} - R_{S,B}\right) \times W_{S,B}\right)\right]$$

using:

$$W_{S,B} = \sum_{g=1}^{n} W_g$$

$$R_{S,P} = \sum_{g=1}^{n} C_g$$

$$R_{S,B} = \sum_{j=1}^{n} C_j$$

where:

S_S: Selection effect for sector **S** in portfolio or parent segment **P**

$W_{S,B}$: Weight of segment **S** in relation to benchmark or parent segment **B** as a whole in percent

$R_{S,P}$: Absolute contribution of segment **S** in the portfolio or parent segment **P** to the portfolio or parent **P** as a whole

$R_{S,B}$: Absolute contribution of segment **S** in the benchmark or parent segment **B** to the benchmark or parent segment **B** as a whole

n: Number of instruments in a segment, portfolio, or benchmark, respectively, when analysing a segment, portfolio, or benchmark

Both, the allocation and selection effects account for only part of the total differential return between benchmark and portfolio. The residual, is explained by the Residual or Interaction Effect, which is driven by both, the asset allocation and security selection decisions interaction together. The details for calculating the Interaction Effect are set out in Formula 8-21.

Formula 8-21: Interaction effect

$$I_S = \left[\left(R_{S,P} - R_{S,B}\right) \times \left(W_{S,P} - W_{S,B}\right)\right]$$

using:

$$W_{S,B} = \sum_{g=1}^{n} W_g$$

$$R_{S,P} = \sum_{g=1}^{n} C_g$$

$$W_{S,j} = \sum_{g=1}^{n} W_j$$

$$R_{S,B} = \sum_{j=1}^{n} C_j$$

where:

I_S: Interaction effect for sector **S** in portfolio or parent segment **P**

$W_{S,P}$: Weight of segment **S** in relation to portfolio or parent segment **P** as a whole in percent

$W_{S,B}$: Weight of segment **S** in relation to benchmark or parent segment **B** as a whole in percent

$R_{S,P}$: Absolute contribution of segment **S** in the portfolio or parent segment **P** to the portfolio or parent **P** as a whole

$R_{S,B}$: Absolute contribution of segment **S** in the benchmark or parent segment **B** to the benchmark or parent segment **B** as a whole

n: Number of instruments in a segment, portfolio, or benchmark, respectively, when analysing a segment, portfolio, or benchmark

To arrive at the Total Attribution Effect you need to add all three—allocation, selection, and interaction—effects together as set out in Formula 8-22.

Formula 8-22: Attribution total effect for a segment

$$T_S = \left[A_S + S_S + I_S\right]$$

where:

T_S: Total effect for sector **S** in portfolio or parent segment **P**

A_S: Allocation effect for sector **S** in portfolio or parent segment **P**

S_S: Selection effect for sector **S** in portfolio or parent segment **P**

I_S: Interaction effect for sector **S** in portfolio or parent segment **P**

BHB attribution is quite an involved method. You will therefore benefit by not just using the supplied portfolio data from the lab exercise, but also by playing a bit with the data. Load different portfolio composition and price data scenarios and then using the report to explore the difference in results.

Lab Exercise 8.7: BHB Performance Attribution

1. Open the file **BHBPerformanceAttrib.sql** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the ORACLE PL/SQL code illustrating the performance calculation shown above.
2. Using Oracle run the following examples at your Oracle SQL Plus command line using the example database for this book you created from the scripts downloaded from the companion web site:

```
SQL> @BHBPerformanceAttrib.sql

...

SQL> InitialiseBHBPerformancetAttrib ()

...

SQL> RunBHBPerformanceAttrib ()

...

SQL> BHBPerformanceReport()
```

8.4 Performance Using Risk Adjusted Return on Capital

So far in this chapter we have concentrated on performance measures for portfolios of financial instruments. However, as highlighted at the beginning of the chapter, many such portfolios are managed in institutions like banks and insurance companies, where it is useful to compare their performance with non-portfolio-based business activities. The types of measures needed for such a comparison is usually one of the variants of Risk Adjusted Return On Capital.

8.4.1 RAROC

The first variant to look at is RAROC, Risk Adjusted Return on Capital itself. As shown in Figure 8.8 this is the ratio of risk adjusted return to economic capital employed. To calculate RAROC you can use Formula 8-23. In order obtain the required risk adjusted capital or estimate of expected losses you will need to use either the VAR or, preferably, ETL methods applied to the activity or portfolio concerned. Refer back to Chapters 6 and 7 for more details on VAR and ETL.

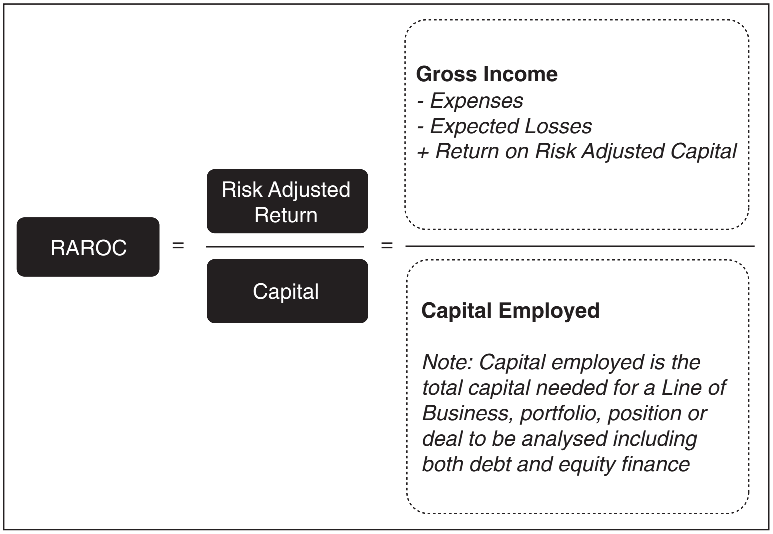


Figure 8.8: The components of RAROC.

Formula 8-23: RAROC risk adjusted return on capital

$$\begin{aligned} R_{RARORAC} &= \text{Risk Adjusted Return/Capital} \\ &= R_{Risk\ Adjusted} / C_{Unadjusted} \\ &= \left[-e - E(L) + Y\left(C_{Risk\ Adjusted}\right)\right] / C_{Unadjusted} \end{aligned}$$

$$R_{Risk\ Adjusted} = i - e - E(L) + Y\left(C_{Risk\ Adjusted}\right)$$

where:

$R_{Risk\ Adjusted}$: Risk adjusted return for the line of business or portfolio

$C_{Unadjusted}$: Total capital needed for a line of business, portfolio, position, or deal to be analysed, including both debt and equity finance

i: Gross income from the line of business or portfolio

e: Expenses for the line of business or portfolio

E(L): Expected losses for the line of business or portfolio

$Y(C_{Risk\ Adjusted})$: Return from required risk adjusted capital

8.4.2 RORAC

The next variant of risk adjusted return on capital is RORAC, or Return on Risk Adjusted Capital. As shown in Figure 8.9 this is really the return on risk-adjusted capital. The details for calculating RORAC are set out in Formula 8-24.

Formula 8-24: RORAC return on risk adjusted capital

$$\begin{aligned} R_{RORAC} &= \text{Return/Risk Adjusted Capital} \\ &= R_{Unadjusted} / C_{Risk\ Adjusted} \\ &= \left[-e - E(L) + Y\left(C_{Risk\ Adjusted}\right)\right] / [ETL_{LOB\ or\ PTF}] \end{aligned}$$

$$R_{Unadjusted} = i - e$$

$$C_{Risk\ Adjusted} = ETL_{LOB\ or\ PTF} \ \textit{or}\ VAR_{LOB\ or\ PTF}$$

where:

$R_{Unadjusted}$: Return for the line of business or portfolio

$C_{Risk\ Adjusted}$: Risk adjusted capital required; ideally this should be ETL but alternatively could be VAR

i: Gross income from the line of business or portfolio

e: Expenses for the line of business or portfolio

E(L): Expected losses for the line of business or portfolio

$Y(C_{Risk\ Adjusted})$: Return from required risk adjusted capital

$ETL_{LOB\ or\ PTF}$: Expected tail loss calculated to a level of confidence that would assure ongoing survival of the business.

$VAR_{LOB\ or\ PTF}$: Expected tail loss calculated to a level of confidence that would assure ongoing survival of the business

8.4.3 RARORAC

Both RAROC and RORAC are only half-way house measures of risk adjusted return on capital since one adjusts only the capital and the other only the return. The combined method that adjusts both capital and returns for risk is called RARORAC, or Risk Adjusted Return On Risk Adjusted Capital. The components of RARORAC are illustrated in Figure 8.10 and the calculation details for RARORAC are covered in Formula 8-25.

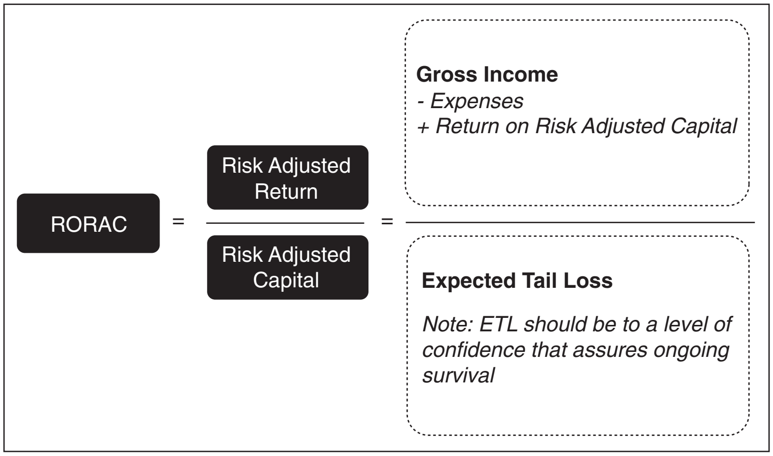


Figure 8.9: The components of RORAC.

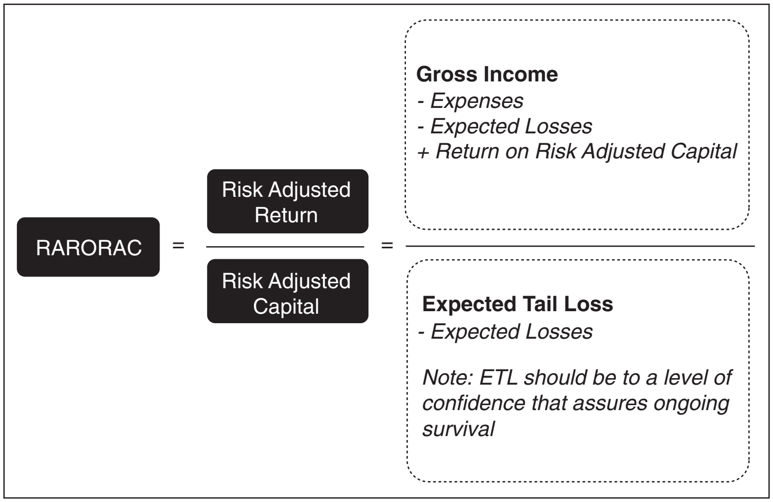


Figure 8.10: The components of RARORAC.

Formula 8-25: RARORAC risk adjusted return on risk adjusted capital

$$
\begin{aligned}
R_{RARORAC} &= \text{Risk Adjusted Return/Risk Adjusted Capital} \\
&= R_{Risk\ Adjusted} / C_{Risk\ Adjusted} \\
&= \left[-e - E(L) + Y\left(C_{Risk\ Adjusted}\right)\right] / \left[ETL_{LOB\ or\ PTF}\right]
\end{aligned}
$$

$$R_{Risk\ Adjusted} = i - e - E(L) + Y\left(C_{Risk\ Adjusted}\right)$$

$$C_{Risk\ Adjusted} = ETL_{LOB\ or\ PTF} \text{ or } VAR_{LOB\ or\ PTF}$$

where:

$R_{Risk\ Adjusted}$: Risk adjusted return for the line of business or portfolio

$C_{Risk\ Adjusted}$: Risk adjusted capital required; ideally this should be ETL but alternatively could be VAR

i: Gross income from the line of business or portfolio

e: Expenses for the line of business or portfolio

E(L): Expected losses for the line of business or portfolio

$Y(C_{Risk\ Adjusted})$: Return from required risk-adjusted capital

$ETL_{LOB\ or\ PTF}$: Expected tail loss calculated to a level of confidence that would assure ongoing survival of the business.

$VAR_{LOB\ or\ PTF}$: Expected tail loss calculated to a level of confidence that would assure ongoing survival of the business

The RAROC lab in Lab Exercise 8.8 will use prepared sample data so you will not need to generate VAR or ETL as inputs to the lab. It covers all three measures—RAROC, RORAC, and RARORAC. Once you are familiar with both the RAROC lab and the VAR and ETL lab exercises you could try and combine them by first calculating VAR or ETL figures for several portfolios or activities and then using the RAROC lab tools to compare them.

Lab Exercise 8.8: RAROC Performance

1. Open the file **RAROCPerformance.sql** from the set of sample files from the companion web site for this book (http://modelbook.fincorefinancial.com/) and explore the ORACLE PL/SQL code illustrating the performance calculation shown above.
2. Using Oracle run the following examples at your Oracle SQL Plus command line using the example database for this book you created from the scripts downloaded from the companion web site:

```
SQL> @RAROCPerformance.sql

...

SQL> Initialise RAROCPerformance()

...

SQL> RunRAROCPerformance ()

...

SQL> RAROCPerformanceReport()
```

8.5 Investment Performance Management

The final part of this chapter goes back to portfolios again. As set out in the beginning, the information from performance and attribution feeds into a continuous iterative cycle for portfolio management, and should help optimise the portfolio performance going forward.

To achieve an optimal portfolio allocation at both an asset class and stock level it is normally advisable to use optimisation. The classical method proposed by Harry Markowitz is to use Quadratic programming to solve the problem as he formulated it.

However with the use of Monte Carlo simulation, there is a relatively simple alternative. If you can simulate a sufficiently large set of alternative outcomes for each asset, the problem can be formulated as a linear program with the objective of maximising the portfolio return with a suitable combination of weights for the asset, which also satisfies the constraint that the maximum loss should not exceed a certain level given by the portfolio manager.

This can be formulated as the Linear Program shown in Formula 8-25.

Formula 8-26: Linear program for portfolio optimisation using simulated returns

$\textit{Maximise} \sum_{i=1}^{n} w_i \times x_i$

subject to:

$\sum_{j=1}^{n} w_{1,j} \times x_{1j} = y_2$

$\sum_{j=1}^{n} w_{2,j} \times x_{2j} = y_1$

...

$\sum_{j=1}^{n} w_{m-1,j} \times x_{m-1,j} = y_{m-1}$

$\sum_{j=1}^{n} w_{m,j} \times x_{m,j} = y_m$

where:

$\textit{Maximise} \sum_{i=1}^{n} w_i \times x_i$ **:** Is the value of the portfolio to be maximised and w_i is the weight to be determined for the ith instrument in the portfolio, and x_i is the **i**th instrument

$\sum_{j=1}^{n} w_{m,j} \times x_{n,j} = y_m$ **:** Is the **m**th constraint for **m** in **1 … m**. The constant y_m should be set to the portfolio manager's maximum loss and the $w_{m,j}$ should be set to the simulated return for instrument j in the **m**th scenario

If you are not familiar with optimisation by linear programming the best is to work through the lab exercise and the experiment, modifying the asset returns in the simulated scenario and varying the portfolio manager's maximum allowed loss.

Lab Exercise 8.9: LP Portfolio Optimisation

1. Open the file **LPPortfolioOptimisation.R** from the set of sample files from the companion web site for this book (http://modelbook.bancstreet.com/) and explore the R code illustrating the performance calculation shown above.

2. Using the R application load the file **LPPortfolioOptimisation.R** and run the following examples at the command prompt in your R GUI:

```
R> InitialiseLPPTFOptimise()

...

R> RunLPPTFOptimise ()

...

R> LPPTFOptimiseReport()
```

CHAPTER 9

Implementing Performance Models

In this chapter, we demonstrate step by step how to implement performance measurement processes in an evolutionary, iterative way using the same implementation framework as in the previous chapters. We show how to

- define and use checklists that drive and support performance measurement implementation,
- store and manage the meta-, input, and output data vital for performance measurement within the framework of the data model blueprint introduced in Chapter 2.

9.1 Performance Measurement Implementation: Step by Step

There are four main steps in a performance model implementation using the building block approach. We distinguish five main steps, depicted in Figure 9.1, that will be discussed at full length in the subsequent sections.

1. Compose the Performance Measurement Model Structure: Decide how to decompose the performance of any set of portfolios that you want to measure as a composite. This will allow you to use hierarchical attribution of performance layer by layer, as discussed in Chapter 8. You will do this by grouping the individual portfolios into successively higher level composites in a hierarchy until you reach the top-level composite in your hierarchy, which covers all portfolios in your composite hierarchy.
2. Define Performance Benchmarks: You will need to map a benchmark to each base portfolio and composite portfolio in the overall composite hierarchy you created in step one.
3. Map Benchmarks to Portfolios: You will need to build and calibrate the performance benchmarks you have mapped to your portfolios in the composite hierarchy.
4. Create Performance Measurement Process: Create an overall process that runs all the different steps needed to perform the end-to-end performance calculations, and to write the results back into the database so that they can later be used from online performance dashboards, in reports or feeds to other systems.

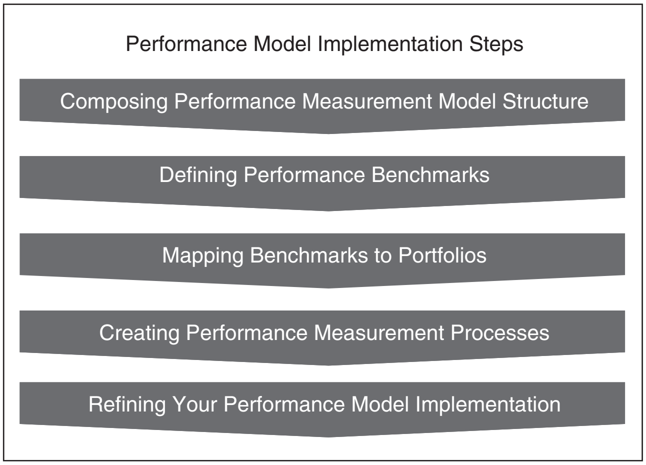


Figure 9.1: Performance measurement implementation steps.

5. Refine the Risk Model Implementation: The best way to start is to begin the implementation with just a single portfolio and no hierarchy at first, and then when this single level performance analysis works to your satisfaction, move on to multiple portfolios grouped into one composite, and only after that works, move on to implementing multilevel hierarchies.

We will now look at each step in turn and then briefly consider how best to shape the gradual refinement of your performance measurement implementation.

9.1.1 Composing the Performance Portfolio Model

The first step in your performance measurement implementation is to decide how to decompose the performance of any set of portfolios that you want to measure as a composite. The decomposition of the performance impact at different layers of the management of the portfolios in the composite will allow you to use the tools of hierarchical attribution introduced in Chapter 8.

Performance attribution will help you to better understand the contribution made by the different elements of your portfolio management process. To carry out this decomposition you will need to group individual base portfolios into successively higher level composites in a hierarchy until you reach the top-level composite in your hierarchy, which covers all portfolios in your composite hierarchy.

9.1.1.1 Base Portfolios and Positions

Figure 9.2 shows a diagrammatic representation of a three-level portfolio hierarchy with individual base portfolios at the bottom, each with the individual holdings and two levels of

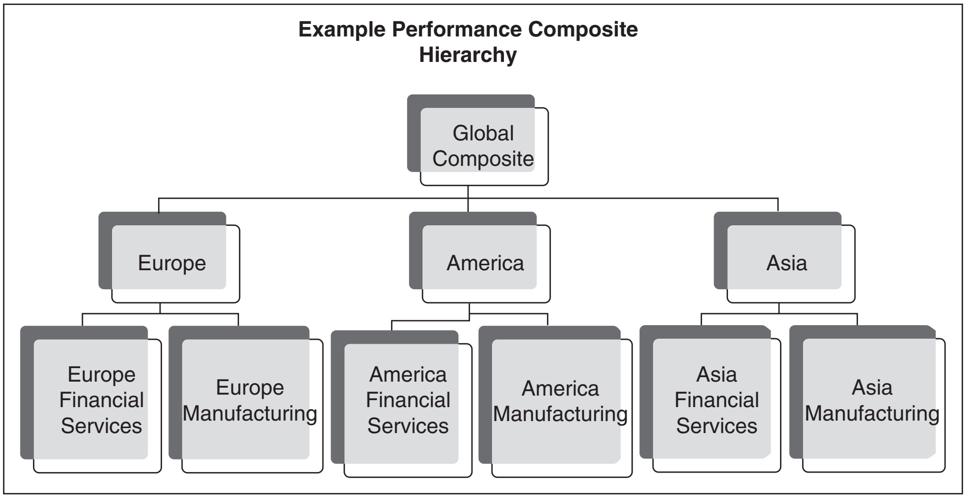


Figure 9.2: Example of a three-level composite hierarchy.

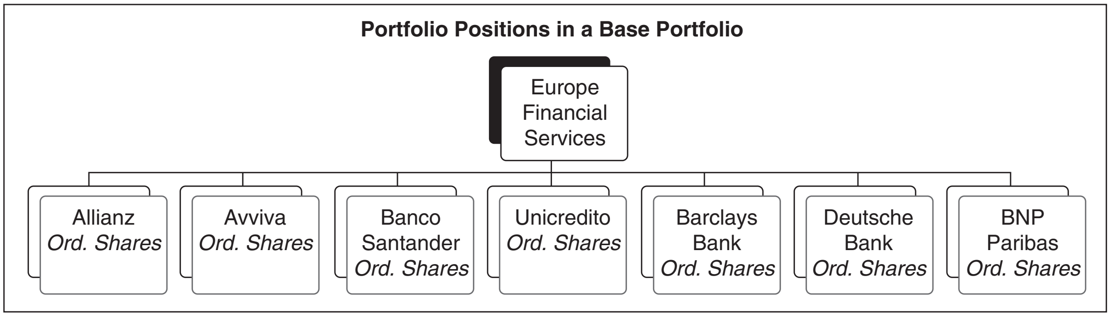


Figure 9.3: Example of portfolio positions in a base portfolio.

composite portfolios successively aggregating the base portfolios. In the example the base portfolios each are containing the holdings for stocks subdivided by both region and sector. Each base portfolio thus contains only holdings from one region and one sector.

For example the base portfolio EUROPE—FINANCIAL SERVICES should contain only holdings of financial instruments providing exposure to the region "Europe" and the sector "Financial Services." In Figure 9.3 you can see the base portfolio EUROPE—FINANCIAL SERVICES expanded to show the holdings in the portfolio in the form of a tree diagram.

Each leaf in the tree of Figure 9.3 will be represented by a position record in the database similar to the record structure in Figure 9.4. The same record structure can also be used to record the valuation at a given point in time such as the beginning or the end of a

Portfolio Position			
CASHFLOW ELEMENT ATTRIBUTE	TYPE	LENGTH	MAPPING or CONSTANT VALUE
Portfolio Instrument	NUMBER		ID of the portfolio.
Portfolio Version	VARCHAR2	32	ID of the portfolio snapshot or version.
Constituent Instrument	NUMBER		ID of the instrument held in the portfolio.
Quantity	NUMBER	22	Long (+) or short (–) position held.
Valuation	NUMBER	22	This is either the cost, valuation, or period return for the position.
Effective Date	DATE	7	Date of last revision to the position.
Position Transaction Reference	NUMBER		Cross reference to the transaction that established the position.
Position Function	VARCHAR2	32	Type of position
Position Stage	VARCHAR2	32	Type of position
Currency	VARCHAR2	3	Currency of the position.
Is Valid	NUMBER	22	Indicates whether the position is still in use for current or historical reporting or has been entirely superseded.
Valid From Date	DATE	7	Date from which the position existed or the beginning of a period from which to measure the return on the position.
Valid To Date	DATE	7	Last date on which the position existed or the end of a period from which to measure the return on the position.

Figure 9.4: Structure for tracking portfolio position records, valuations, and returns.

performance period. Finally it can also be used to record the return in money and percentage terms between two dates.

In Figure 9.5 we show an example of a running position record for a position in a given instrument. This is the starting point for valuations and all subsequent steps in performance measurement. In order to calculate the return over a period we need to know at least the position at both the beginning and the end of the period. Ideally, we measure the return at least each time any position in the given portfolio changes in size as well as at any point in time that is the beginning or end of a performance measurement period.

Portfolio Position			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**EXAMPLE VALUE**
Portfolio Instrument	NUMBER		0000011
Portfolio Version	VARCHAR2	32	CURNETPOS
Constituent Instrument	NUMBER		543789 UBS AG Ordinary Shares
Quantity	NUMBER	22	434,000
Original Cost	NUMBER	22	458,789
Effective Date	DATE	7	15/03/2008
Position Transaction Reference	NUMBER		2300990102
Position Function	VARCHAR2	32	Running Position
Position Stage	VARCHAR2	32	4 - Settled
Currency	VARCHAR2	3	CHF
Is Valid	NUMBER	22	True
Valid From Date	DATE	7	23/09/2007
Valid To Date	DATE	7	

Figure 9.5: Example of a portfolio position record in a base portfolio.

In order to obtain a valuation, we multiply the size of the position with the current price at the date of the valuation to obtain the total current value of the position for the valuation date. We can store this information again in a new record in the same structure but using a different value for the position function indicator field.

Figure 9.6 shows an example of such a valuation record derived from the running position in Figure 9.5.

Once we have two valuations, one for the beginning and one for the end of the period, we can also calculate the period return. The resulting position period return can then be stored again in the position structure. In Figure 9.7 you can see an example of how the period return can be recorded.

In order to perform performance attribution, and also for many reports, we need the relative size of a position in percentage terms compared to the portfolio in which it is placed. Figure 9.8 shows how this can be achieved again using the standard portfolio position structure.

9.1.1.2 Portfolio Composites

The higher level composite portfolios in the portfolio hierarchy like the ones shown in Figure 9.2 can again be built using the same machinery as the base portfolio. Each composite portfolio thus again is set up as a portfolio instrument itself. The composite

Portfolio Position Valuation			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**EXAMPLE VALUE**
Portfolio Instrument	NUMBER		0000011
Portfolio Version	VARCHAR2	32	VAL-2008-03-12
Constituent Instrument	NUMBER		543789 UBS AG Ordinary Shares
Quantity	NUMBER	22	434,000
Valuation	NUMBER	22	635,903
Effective Date	DATE	7	15/03/2008
Position Transaction Reference	NUMBER		2300990102
Position Function	VARCHAR2	32	Current Valuation
Position Stage	VARCHAR2	32	4 - Settled
Currency	VARCHAR2	3	CHF
Is Valid	NUMBER	22	True
Valid From Date	DATE	7	15/03/2008
Valid To Date	DATE	7	15/03/2008

Figure 9.6: Example of a valuation of a portfolio position in a base portfolio.

Portfolio Position Return			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**EXAMPLE VALUE**
Portfolio Instrument	NUMBER		0000011
Portfolio Version	VARCHAR2	32	VAL-2008-03-12
Constituent Instrument	NUMBER		543789 UBS AG Ordinary Shares
Quantity	NUMBER	22	5.4%
Valuation	NUMBER	22	26,899
Effective Date	DATE	7	11/03/2008
Position Transaction Reference	NUMBER		2300990552
Position Function	VARCHAR2	32	Period Return
Position Stage	VARCHAR2	32	4 - Settled
Currency	VARCHAR2	3	CHF
Is Valid	NUMBER	22	True
Valid From Date	DATE	7	15/02/2008
Valid To Date	DATE	7	15/03/2008

Figure 9.7: Example of a record of the return on a position over a one-month period.

Portfolio Position as Percentage of Portfolio			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**Example Value**
Portfolio Instrument	NUMBER		0000011
Portfolio Version	VARCHAR2	32	VAL-2008-03-12
Constituent Instrument	NUMBER		543789 UBS AG Ordinary Shares
Quantity	NUMBER	22	12.5%
Valuation	NUMBER	22	NULL
Effective Date	DATE	7	11/03/2008
Position Transaction Reference	NUMBER		2300990552
Position Function	VARCHAR2	32	Percentage of Portfolio
Position Stage	VARCHAR2	32	4 - Settled
Currency	VARCHAR2	3	CHF
Is Valid	NUMBER	22	True
Valid From Date	DATE	7	15/02/2008
Valid To Date	DATE	7	15/03/2008

Figure 9.8: Example of a record of the relative size on a position in a (base) portfolio.

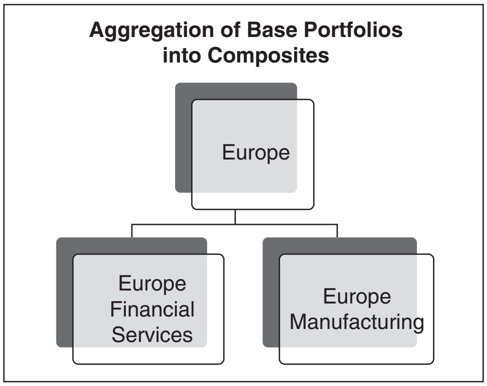


Figure 9.9: Aggregation of base portfolios into composite portfolios.

portfolio Europe in Figure 9.9 can thus be defined as a portfolio instrument, and the two base portfolios, Europe—Financial Services and Europe—Manufacturing can be linked to the composite via a special running position record for each.

An example running position record linking the base portfolio Europe—Financial Services to the composite portfolio Europe is shown in Figure 9.11. The Global Composite portfolio in Figure 9.10 can yet again be built the same way as the composite portfolio Europe.

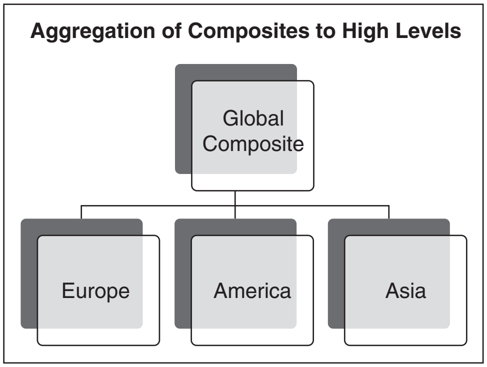


Figure 9.10: Aggregation of composite portfolios to higher level composites.

Portfolio Positions			
CASHFLOW ELEMENT ATTRIBUTE	TYPE	LENGTH	EXAMPLE VALUE
Portfolio Instrument	NUMBER		0000025 EUROPE
Portfolio Version	VARCHAR2	32	PTFHIER
Constituent Instrument	NUMBER		0000011 EUROPE – Fin. Services
Quantity	NUMBER	22	1
Valuation	NUMBER	22	
Effective Date	DATE	7	15/03/2008
Position Transaction Reference	NUMBER		2300990102
Position Function	VARCHAR2	32	Portfolio Composite
Position Stage	VARCHAR2	32	4 - Settled
Currency	VARCHAR2	3	CHF
Is Valid	NUMBER	22	True
Valid From Date	DATE	7	23/09/2007
Valid To Date	DATE	7	

Figure 9.11: Example position record linking a portfolio to a higher level composite.

The Global Composite portfolio should thus be set up as a separate portfolio instrument. Then, each of the component composites, Europe, America, and Asia, can be linked to the Global Composite Portfolio instrument using a portfolio position record similar to the one in Figure 9.11.

Once the portfolio hierarchy is completely set up and the valuations, returns, and relative composition of the base portfolios have been calculated, it is possible to calculate

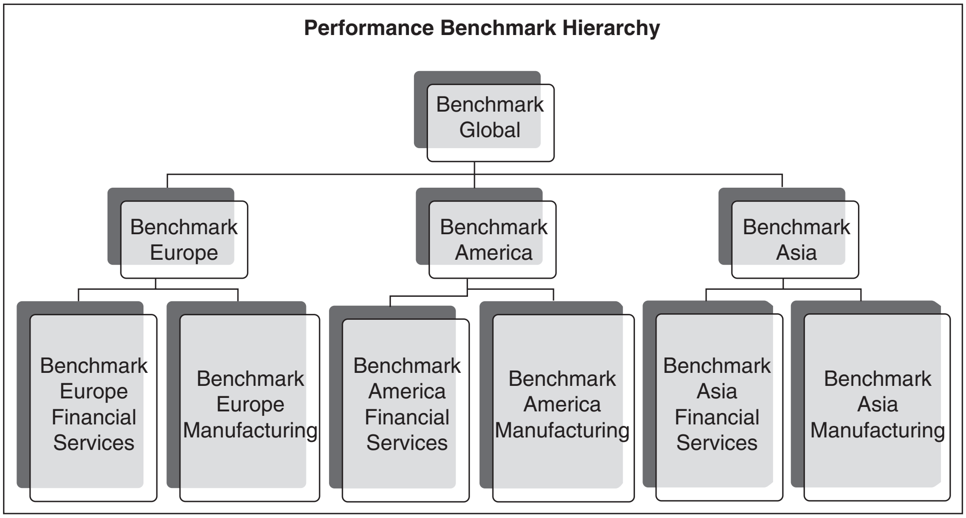


Figure 9.12: An example of a hierarchical benchmark structure.

aggregated valuations, returns, and relative composition for each composite from the base portfolios upward. With these calculations for each composite in your hierarchy you have the key required inputs from a portfolio perspective for the hierarchical attribution calculations as those in Chapter 8. In the next section we look at how to create the equivalent information for benchmarks.

9.1.2 Defining Performance Benchmarks

Although we can measure the absolute performance of portfolios with the portfolio structures shown in the previous section, we need some more information to analyse the performance of portfolios relative to a benchmark and to carry out attribution analysis that is based on such relative performance measures.

To do both, relative performance and attribution analysis, we will need to define a matching benchmark hierarchy that can be compared with the actual portfolio. Figure 9.12 provides a graphical representation of the benchmark hierarchy equivalent to the portfolio composite hierarchy from Figure 9.2.

The composition of each benchmark is recorded using the benchmark equivalent of the portfolio component. Figure 9.13 shows the generic description of the attributes of the benchmark component record structure. It allows the recording of the composition of each benchmark from component positions in other financial instruments or lower level benchmark components.

Benchmark Component			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Benchmark Instrument	NUMBER		Id of the Benchmark
Benchmark Version	VARCHAR2	255	Id of the Benchmark Snapshot or Version
Component Instrument	NUMBER		Id of the Instrument held in the benchmark
Weight Factor	NUMBER	22	Weight of the component in the benchmark
Return Factor	NUMBER	22	Return of the component over a given period
Chaining Factor	NUMBER	22	
Effective Date	DATE	7	Date of last revision to the position.
Position Function	VARCHAR2	32	Type of position
Position Stage	VARCHAR2	32	Type of position
Currency	VARCHAR2	3	Currency of the
Is Valid	Is Valid	NUMBER	Indicates whether the position is still in use for current or historical reporting or has been entirely superseded
Valid From Date	Valid From Date	DATE	Date from which the position existed or the beginning of a period for which to measure the return on the position.
Valid To Date	Valid To Date	DATE	Last date on which the position existed or the end of a period for which to measure the return on the position.

Figure 9.13: Description of a benchmark component record.

Figure 9.14 provides an example of a component position at the base level of the benchmark hierarchy. In this example it records that the total value of the Benchmark component Europe—Financial Services is composed of 8.5 percent of UBS Ordinary Shares.

Figure 9.15 shows how lower level benchmark components can be linked into higher level benchmark components. The example is that the total value of the Benchmark component Europe is composed of 55 percent of the benchmark component Europe—Financial Services.

Benchmark Component			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Benchmark Instrument	NUMBER		000000153 – Bench. Europe Fin. Serv.
Benchmark Version	VARCHAR2	255	CurBmkCmp
Component Instrument	NUMBER		543789 UBS AG Ordinary Shares
Weight Factor	NUMBER	22	8.50%
Return Factor	NUMBER	22	5.4%
Chaining Factor	NUMBER	22	NULL
Effective Date	DATE	7	5/03/2008
Position Function	VARCHAR2	32	Type of position
Currency	VARCHAR2	3	CHF
Is Valid	Is Valid	NUMBER	True
Valid From Date	Valid From Date	DATE	15/02/2008
Valid To Date	Valid To Date	DATE	15/03/2008

Figure 9.14: An example of a portfolio position record in a base portfolio.

Benchmark Component			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**MAPPING or CONSTANT VALUE**
Benchmark Instrument	NUMBER		000000167 – Bench. Europe
Benchmark Version	VARCHAR2	255	CurBmkCmp
Component Instrument	NUMBER		000000153 – Bench. Europe Fin. Serv
Weight Factor	NUMBER	22	55.00%
Return Factor	NUMBER	22	6.73%
Chaining Factor	NUMBER	22	NULL
Effective Date	DATE	7	5/03/2008
Position Function	VARCHAR2	32	Type of position
Currency	VARCHAR2	3	GBP
Is Valid	Is Valid	NUMBER	True
Valid From Date	Valid From Date	DATE	15/02/2008
Valid To Date	Valid To Date	DATE	15/03/2008

Figure 9.15: An example of a portfolio position record in a base portfolio.

9.1.3 Mapping Benchmarks to Portfolios

Now that we have created both, a composite portfolio hierarchy and a matching benchmark hierarchy, we need to be able to map one to the other. The structure that allows us to do this is the Instrument Relationship Condition.

Figure 9.16 shows a description of the attributes of the Instrument Relationship Condition. Figure 9.17 gives an example of the structure in use. Each instance like the one in Figure 9.17 links one portfolio instrument to its matching benchmark.

9.1.4 Creating Performance Measurement Processes

Once you have your reference and market data in your database and correctly set up all portfolios and benchmarks and their mappings, you will need to orchestrate the whole end-to-end process. The end-to-end process usually comprises a number of sub-processes, all of which need to be coordinated. In Figure 9.18 you can see a typical two stage process sequence. This process takes portfolio and benchmark reference data and market data, as in a first stage performance measurement process, through to performance results. The performance results are then fed into a second stage performance attribution, creating the performance attribution analysis.

Instrument Relationship Condition			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**Description**
Instrument	NUMBER		ID of the portfolio
Condition Scheme	VARCHAR2	32	Group of relationship functions
Condition Scheme Function	VARCHAR2	32	Relationship function
Condition Relationship Type	VARCHAR2	32	Type of relationship
Condition Instrument	NUMBER		Id of the mapped benchmark
Condition Instrument Role Type	VARCHAR2	32	Role of the mapped benchmark
Condition Minimum	NUMBER	22	
Condition Maximum	NUMBER	22	
Constraint Type	VARCHAR2	32	
Is Valid	NUMBER	22	Indicates whether the position is still in use for current or historical reporting or has been entirely superseded
Valid From	DATE	7	Last date on which the position existed
Valid To	DATE	7	Date from which the position existed

Figure 9.16: Description of the relationship condition structure.

Portfolio Relationship Condition			
CASHFLOW ELEMENT ATTRIBUTE	**TYPE**	**LENGTH**	**DESCRIPTION**
Instrument	NUMBER		00000011
Condition Scheme	VARCHAR2	32	PERFORMANCE
Condition Scheme Function	VARCHAR2	32	Benchmark mapping
Condition Relationship Type	VARCHAR2	32	Mapping
Condition Instrument	NUMBER		000000153
Condition Instrument Role Type	VARCHAR2	32	
Condition Minimum	NUMBER	22	
Condition Maximum	NUMBER	22	
Constraint Type	VARCHAR2	32	
Is Valid	NUMBER	22	True
Valid From	DATE	7	17/01/2008
Valid To	DATE	7	

Figure 9.17: An example of the use of the relationship condition structure.

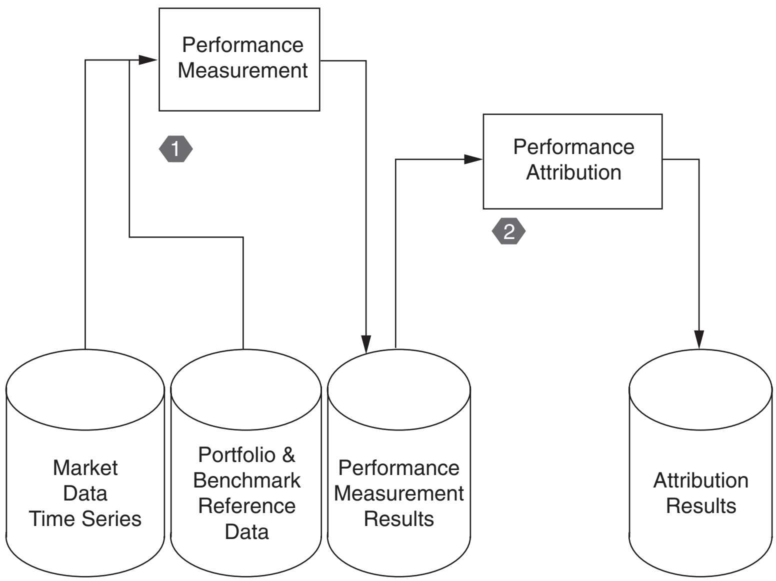


Figure 9.18: End-to-end process for performance measurement and attribution.

Although it is possible to do the coordination of the end-to-end performance process using hardcoded program code, the best way to achieve this is by using a process orchestration framework such as the one introduced in Chapter 5 for orchestrating instrument pricing processes. In fact, the same structures will work just as well for performance measurement.

9.1.5 Refining Your Performance Measurement Implementation

Performance measurement involves large numbers of assumptions and judgment calls that have to be made well before a result becomes visible. It is inevitable that at least some assumptions and judgment calls will have to be revised. The fewer assumptions and judgment calls involved, and the quicker you can get to the point where you can check them against real results, the easier it will be to identify what needs to be changed and to put the revisions into practice.

The fountain model for model implementations that is reproduced in Figure 9.19 provides a robust framework that will help you make sure that frequent small iterations do not veer off track.

In practical terms this means that you should start your implementation with a single-level portfolio performance measurement process.

The first iteration will be a solution that you can already put into day-to-day production even though it may be somewhat basic. You will gain much from the experience of having your solution in production as this experience will provide you very valuable feedback and guidance to help shape future iterations.

Many issues will not become visible until your solution goes live, and the earlier in the life of a solution that you discover a problem the easier and cheaper it will be to fix it.

Small Iterations as the Road to Success

- Start with a single level performance measurement model.
- Build your hierarchy to cover one more level each time you have the simpler model working well in practice.
- Focus on getting valuations and measurement of returns until your reliability of those is impeccable.
- Early iterations should be as short as possible, try one week iterations if at all possible.
- Focus on modelling rather than infrastructure – if possible use a purpose designed model implementation platform such as Fincore Financial’s Analytics Hub or other similar solution.

Figure 9.19: Small iterations as the road to success.

Finally, whenever you can, try to avoid getting bogged down in infrastructure issues unless you want to give up model development and focus on infrastructure instead. You and your team will never have enough time to do full justice to both, and the earlier you make that decision the better.

9.2 Performance Measurement Implementation Checklists

In Chapter 1 we introduced a framework for model implementations. This is summarised by the charts in Figures 9.20, 9.21, and 9.22. All three charts are meant as checklists rather than rigid programs of mandatory tasks and deliverables that have to be complete in each iteration.

You may find that, for instance, the Governance and Oversight Perspective as well as the Business Ownership perspective may receive significant attention in the first one or two iterations so that a suitable foundation for future iterations is put together. Following iterations will be much lighter in terms of effort focused on task in these two perspectives since the fruits of this early effort can be used again and again in later iterations.

At some point, when you are about to shift gears from early prototypes and pilots to having your solution in full-scale production, the Governance and Oversight perspective as well as the Business Ownership perspective will come into the foreground again as the nature of your work is shifting and you will need to revise and extend the ground rules you had agreed to earlier with your stakeholders and business sponsors.

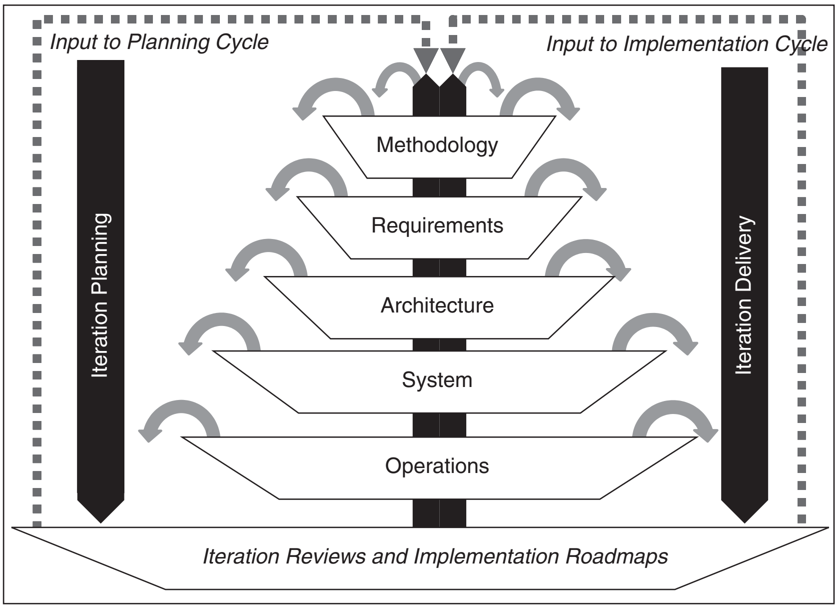


Figure 9.20: Iterative refinement of model implementations.

Model Implementation Framework							
Perspective	Goals Dimension	Organisation Dimension	Schedule Dimension	Model Dimension	Data Dimension	Process Dimension	Role
Methodology Perspective	Goal Setting Guidelines	Organisation Guidelines	Schedule Guidelines	Modelling Guidelines	Data Guidelines	Process Guidelines	Business Governance
Business Perspective	Business Case/ Plan	Organisation Requirements	Schedule Requirements	Model Requirements	Data Requirements	Process Requirements	Business Owner
Architecture Perspective	Architecture Plan	Organisation Specifications	Design Schedule	Model Specifications	Data Specifications	Process Specifications	Model Designer
Systems Perspective	Build Plan	Responsibility Model	Build Schedule	Model Designs	Msg & Data Store Design	Process Designs	Model Builder
Operating Perspective	Operating Plan & Targets	Responsibility Assignments	Maintenance Schedule	Production Models	Production Data Stores	Operating Schedules	Model Operator
	Why?	Who?	When?	How?	What?	Where?	

Figure 9.21: Model implementation framework.

Model Implementation Methodology Template							
Set of Deliverables	Goals	Organisation	Schedule	Model	Data	Process	Persons Responsible
Methodology	Develop and Implement Goal Setting Guidelines	Develop and Implement Organisation Guidelines	Develop and Implement Scheduling Guidelines	Develop and Implement Modelling Guidelines	Develop and Implement Data Guidelines	Develop and Implement Process Guidelines	Governance Board
Requirements	Develop and Implement Business Case & Plan	Develop and Implement Organisation Requirements	Develop and Implement Schedule Requirements	Develop and Implement Model Requirements	Develop and Implement Data Requirements	Develop and Implement Process Requirements	Business Owner
Architecture	Develop and Implement Architecture Plan	Develop and Implement Organisation Specifications	Develop and Implement Design Schedule	Develop and Implement Model Specifications	Develop and Implement Data Specifications	Develop and Implement Process Specifications	Model Designer
System	Develop and Implement Build Plan	Develop and Implement Responsibility Model	Develop and Implement Build Schedule	Develop and Implement Model Designs	Develop and Implement Msg & Data Store Design	Develop and Implement Process Designs	Model Builder
Operations	Develop and Implement Operating Plan & Targets	Develop and Implement Responsibility Assignments	Develop and Implement Maintenance Schedule	Develop and Implement Production Models	Develop and Implement Production Data Stores	Develop and Implement Operating Schedules	Model Operator
	Why?	Who?	When?	How?	What?	Where?	

Figure 9.22: Model implementation methodology template.

When you start planning for an iteration you and your team can quickly go through the cells in each diagram and make a call of whether the item in the cell needs attention in the forthcoming iteration. Then your planning and work can concentrate on those items that matter for the current iteration.

9.2.1 Methodology Perspective: Performance Governance Oversight

You take again the town planner's view of your performance measurement implementation. It is important to make the ground rules fit the size and nature of your endeavour. If you simply tinker with performance measurement on the side, perhaps to explore their potential or better understand them, then you will need very little groundwork in this perspective.

If, on the other hand, your performance measurement results are used for key financial decisions or will be used as part of a service or product, then the work that has to be undertaken in this perspective will be substantial. We restate the key points that your planning should cover under this perspective:

- Who will use your performance measurement and for what purpose?
- What will happen if your performance measurement results are wrong or unavailable?
- Who would have to answer the questions if things go wrong after this or the next few iterations: you or your team, your boss, the CEO or the Chairman of the Board of your organisation?
- Have you included the right people in the dialogues or workshops for setting out the ground rules?
- Have you done enough work to make the ground rules robust enough for the current stage?
- Have you covered all the different dimensions from goals to process?

9.2.2 Business Perspective: Performance Business Ownership

As the name says, you need to take the perspective of a business owner, who has always two questions on his or her mind:

- How will this make your business more profitable?
- How will this protect your business from failing?

Thus you will need to probe for how your performance measurement results will add value to your business or that of your sponsors or clients. Then you need to check yet again for how your performance measurement process could endanger your business or that of your sponsors or clients if it goes wrong. The key points your planning should cover are:

- Do the performance measurement results and process for creating them fit the intended purpose?
- Does your proposed way of creating performance measurement fully exploit the potential for adding value to our business or that of your sponsors or clients?
- Can you do this more cheaply, faster, with less resources, fewer errors, or in a more informative way?
- What is the biggest pitfall that you can fall into with the chosen approach to performance measurement and how would you get out of it? Would you survive the fall?
- Have you got the resources to see your plan through to completion? Can you see it through beyond completion and all the way to success?

9.2.3 Architecture Perspective: Performance Model Designer

The designer's perspective is dealing with the decisions that set the framework for how your models will be implemented in detail. Some of these decisions may be given quantities such as may be the case if you already have a technical infrastructure for your implementation that you must use.

The challenge for the designer is to shape the remaining points in the framework in such a way that both the model builder's work and the model operator's efforts together will yield the desired results both now and in the distant future. The key points your planning should cover are:

- What is the right infrastructure for our implementation?
- Do you have the right infrastructure, and if not, how can you get it?
- Will your proposed implementation work with your given data feeds and the skills and experience of the people who will operate the performance measurement processes?
- What is the roadmap that will deliver as much value as possible as early as possible?
- Are you using this roadmap? If not, what stops you from using it and how can you remove any such obstacles?

9.2.4 Systems Perspective: Performance Model Builder

The system builder's perspective deals with the shop-floor decision of how to actually code and put together the valuations solution. Here we can rely on a lot of ground work already laid by the other perspectives.

The bigger picture in terms of what we aim to achieve, the pitfalls to avoid, and the constraints given or agreed to should already be clear. Thus we can now focus on the details of the implementation. The key points your planning should cover are:

- Are you implementing the performance measurement process in the best way possible?
- Are there faster or more predictable ways at arriving at the same result?
- Are you handling all the pitfalls that your performance measurement process may face?
- Can and do you double check the results? If not, is this OK under the circumstances? If the answer is yes, would your boss or CEO or your clients or regulators agree? If they would agree now would they still agree if our performance measurement results or process went horribly wrong without anyone spotting it for some time?
- If you detect a problem later or a better method becomes available, how difficult would it be to switch? If it is difficult, what can you do now to make that future switch easier?

9.2.5 Operating Perspective: Performance Model Operator

The operator's perspective is dealing with what it is like to use the solution to produce the actual performance measurements under actual operating conditions rather than what may have been assumed or planned. The key points your planning should cover are:

- Do you fully understand the capabilities and limitations of the solution? If not, where do we lack information and how could you bridge this gap?
- Can you run the solution in such a way as to produce performance measurement results to an agreed standard with the given data feeds and the given expertise of the team?
- If not, what are the problems and what can be done in the short term to overcome them?
- How can you help the implementation team to overcome these challenges in the medium to long term?

- Are you making the best use of the solution? If not, what would you need to do to get there?
- What could go wrong with the way we operate the solution? What is the impact on the business and your clients if in the worst case? Are there ways of mitigating this? Are there ways in which the solution could be operated differently that would result in a more favourable worst case or at least reduce the impact?

9.3 Further Performance Measurement Models

In Chapter 8 we introduced a set of commonly used performance measurement models. There are, of course, many more enhancements in which you may be interested. Although there is no hope of covering all even in a book just on performance measurement, the more models to which you have access, the more you will be able to practice.

We therefore have put together two lab exercises: one will introduce you to more extensions to performance measurement, and one will allow you to explore implementing portfolio optimisation.

Lab Exercise 9.1: Enhancements to Performance Measurement

1. Download the tutorial notes **LAB_9_1_EnhancementsToPerformanceMeasurement.pdf** from the companion web site (http://modelbook.bancstreet.com/) to explore further ways to enhance your performance measurement and attribution analysis.

Lab Exercise 9.2: Implementing Portfolio Optimisation

1. Download the tutorial notes **LAB_9_1_ImplementingPortfolioOptimisation.pdf** from the companion web site to explore some of the additional aspects of representing portfolios to implement practical portfolio optimisation.

CHAPTER 10

Understanding Valuation Theory

We present the key principles and elements of modern valuation theory, which is the basis of modern pricing, risk, and performance models. We demonstrate how modern valuation theory allows for building valuation models for virtually any instrument from a manageable set of atomic valuation building blocks described in our approach.

10.1 The Purpose of Valuation Theory

Asset pricing is about finding an appropriate value, or a "price" for a financial instrument, an asset. We use the term **valuation theory** instead of "asset pricing theory" as it is more generic and also points more in the direction of what the theory actually does.

The theory gives an economic rational for the valuation model, so we can justify the results in front of ourselves and our superiors or clients. Giving an economic rational necessarily means working with an economic model, which in turn implies the usage of assumptions and simplifications.

This also points to a practically very important, though often neglected aspect: We talk about a "value" in the first place and not a "price," because the latter always has a connotation to a market. However, if you have a market price you do not need to value the instrument anymore, right? Just take the market price for the asset and you're home and safe.

Of course, this does not work for the situations when valuation theory is relevant: illiquid (rarely or not-at-all traded) instruments, for which no readily available market price exists.

So we are left to ourselves to find a value for such instruments. The most simple models—or rather, approaches—would try to look out for similar instruments that are traded and hence have a market price and infer from here a valuation to our illiquid case. But, how appropriate or safe is that? Many financial instruments, in particular the illiquid ones, are quite unique, and often have very pricing-relevant aspects, such as risk factors, guarantees or embedded optionality. So, finding a "similar but traded" instrument, or—better—set of instruments, though simple in conception, it is a daunting task when trying to implement it.

There is even a theory that formalises a variant of this idea of linking an instrument to another instrument, the so-called Capital Asset Pricing Model (CAPM),[1] which, in its basic form, stipulates a constant relationship between the relative movement of an instrument price to the relative movement of market portfolio. Although the theory is nice, it is a theory. The practical obstacles of bringing it to life are enormous, starting from finding the right market portfolio to a series of simply unjustifiable assumptions about the behaviour of asset prices.

As a reaction, financial theorists have embarked in ever more complicated econometric pricing models, trying to get rid of restrictive assumptions here and there, but so far with very limited success, to say the least.

We therefore recommend that, before blindly following whatever is taught at universities today, you should sit back for a second and think of what you really need the valuation for.

- Is it because you want to find a true value for a traded instrument to be able to exploit market inefficiencies (i.e., you would buy it, if the market value is below the estimated true value)?
- Is it because you are forced by some sort of regulator to mark-to-market your instrument for risk management purposes? The question here is how useful such a marking-to-market of an illiquid security to a nonexistent market could be. What type of information would you generate here?

What appears obvious is that valuation has always a lot to do with subjective judgement. Choosing a model, selecting the implementation parameters, the relevant question is always: "What is the instrument worth to me or my institution?"

This brings us to another important, but often neglected, dialectic of subjective versus market pricing. Much (if not most) of academic valuation theory fares under the heading of arbitrage pricing, claiming that there is perfect information and competition in financial markets, which themselves are assumed (for the theory to work) to be complete; that is, any type of cash flow can be traded at all types in every possible level of decomposition.

Well, if we learned one thing in the past year, it's that markets are *not* perfect and there is *not* always somebody out there willing to exploit any type of (even implied) inefficiency.

So what works under conditions such as the ones prevailing in the current crisis? In our opinion, the one and only true anchor you have when valuing instruments is their cash flow structure, which means real income and repayment to you. To find a value using (expected) cash flows, they need to be discounted properly to take into account both the time value of

[1] See Cochrane (2005), for a very good introduction and overview to this theory, its shortcomings as well as the relevant literature references to this subject.

money (having the same amount of money today is worth more than having it tomorrow, as in the meantime you could earn interest on it) and the opportunity cost of holding that instrument. An additional value diminishing factor is the probability that the issuer of the instrument is actually not going to pay future cash flows, the probability of issuer default. Discounting is done by constructing so-called discount curves that need to incorporate all three aspects.

The valuation approach using this is called the Discount Cash Flow valuation approach, and it is applicable to all types of instruments. We therefore follow Damodaran, if he claims that Discounted Cash Flow valuation is the king path to valuation: "While discounted cash flow [] valuation is only one of the three ways of approaching valuation [which are DCF; relative and contingent claim valuation] and most valuations done in the real world are relative valuations, it is the foundation on which all other valuation approaches are built." (Damodaran, 2002, p.11)

This statement gives then also the tenor for this chapter in which we present to you that part of modern valuation theory that turns around DCF valuation, giving some economic rational to it, and also enlightens the logic of curve construction and discounting.

10.2 Some Notations and Concepts

Valuation theory is developed in a formal language to which we will introduce you bit by bit. We start here by introducing some key symbols and formal conventions that you used throughout this book.

Mathematical Secretism?

We now enter the realm of formal language where most of asset pricing takes place. Therefore a word of comfort to those not yet intimate to the intricacies of mathematical formalism and concepts is warranted: Mathematics, or formal language, is not a secret language designed to make your life miserable and to expose a potential lack of knowledge—it is to help you!

It is easier to write $y = f(x),\ f'(x) > 0\ \forall_x$ instead of saying "y is a positive function of x over the full domain of x."

Therefore, bear with us through this chapter and try to get used to the most relevant set of formal concepts in the world of finance. In the spirit of Chiang and Wainwright (2005) we will do our best to keep you on board and to explain in detail what needs to be explained and simplify as much as possible what can be simplified, at least when it comes to practical application.

10.2.1 On the Use and Meaningfulness of Models

Throughout this book we presented and worked with a series of economic models. When working with them, you should always keep in mind the following caveats.

Models are never an exact description of what actually happens, but rather logical frameworks that help structure your thinking when interpreting real data. Giving models a mathematical formulation often allows us to derive implications of a certain setting/scenario that are not necessarily visible at first sight. Often we also are able to exclude certain interpretations that are not consistent with the chosen framework.

These implications/exclusions are, however, valid only if the assumptions of the model framework are correct. Correct means that they adequately reflect the driving forces of actors' motivations and interactional logic.

The ultimate test for every model is when it is confronted with real data or tested. Still, not being able to reject a model base on real data does not necessarily mean that it is true. It is simply not (yet) falsified. With Karl Popper we say that there is no true model, but only models or theories that have not yet been falsified.

It is important to recall this fundamental scientific wisdom, as—particularly in practical work—models all too often are taken for granted, providing "the truth." When working with models, you should always be wary of them, taking the rich help they can give you but never really trusting them.

10.2.2 Random Variables

A **random variable** or "statistic" is a concept that allows you to capture the fact that for many aspects in life we have to deal with uncertainty. It represents a conceptual shell for a measure of some aspect of real life that may take on any of a predefined set of possible, mutually exclusive values, its **domain**. The assignment of a concrete value to a previously random variable, the **realisation**, is dependent on a specific event, usually determined by the definition of the random variable. Each of the possible values of a domain may be assigned a probability according to which might become the realisation of the random variable. The distribution of probabilities over possible values is called the **probability distribution** of the random variable. Hence, a random variable may be fully characterised by its domain and the corresponding probability distribution.

Let's make this a bit more lively: Think of the random variable "weather tomorrow" that may take on the values "same as today," "better," or "worse." The event that will assign the value to that random variable is the emergence of "tomorrow," because then we will clearly know how the weather is in relation to "today" (then "yesterday").

This example also nicely shows the importance of the domain definition for the meaningfulness of a random variable, as you equally could have assigned the possible values "rain," "sunshine," and "snow." Although the event in both cases will bring along the information you need to assign a value to the random variable, the transformation from observed weather tomorrow into one of the measures defined by the two domains fundamentally determines the meaning and information content represented by the variable.

The example also illustrates the importance of the mutually exclusive property of the values in the domain. This important specification means that nature at any one time can only take on one state of the random variables domain. Think of a domain for the random variable "weather tomorrow" that consists of the states "rain," "fog," and "sunshine." Even though "sunshine" is well enough a state on its own, the state "fog" may imply additionally that it rains! Hence, the information content of the domain is flawed.

The most common example for a random variable is the value of a die. Here, the domain is clearly one of the numbers {1,2,3,4,5,6} and the event determining which of the numbers will be the value is the throwing of a die. The example usually is used in statistics text books to illustrate random variables but its experimental nature does, by construction, not exhibit the typical problems encountered with random variables in financial practice.

We may say that a lot (if not every) of financial aspects of a financial instrument may (need to) be expressed as a random variable, as even promised cash flows may need be paid out due to defaulting to the counterparty or other unforeseen events.

In finance, typical random variables are the:

- Future price of an asset
- Next cash flow on an asset
- Actual discount factor

Which distribution is the right one for a given random variable is the subject of an academic field of its own: **probability** or **stochastic theory**. In practice, it is often not easy to decide which is the right probability distribution. Except for very few obvious cases, like the rolling of a die or the throwing of a coin, the probability distribution of a random variable is a matter of subjective judgement often assisted by some stochastic (or econometric) modelling of an underlying but unknown process that drives the emergence of this or that value for a random variable. Who could know with certainty with which probability the weather tomorrow will take on this or that form?

This brings us to a fundamental problem when working with random variables in finance. Although we reasonably assume that there exists a true distribution for the concepts we identify, we do not know it and must make an assumption-based approximation. In this book, we made it explicit in the various pricing approaches how they solve this dilemma by which assumptions, and the potential consequences of these, for the exactness of the pricing method.

10.2.3 Time and Its Notation

Intertemporal considerations are key to asset pricing, which turns around relating a series of future (and sometimes past) cash flow events to a current, or reference, date. Hence, it is important to clarify the time concepts and corresponding notation.

Throughout the book, we tried to limit our discussion to discrete random variables, closely linked with the concept of discrete time measurement. The reason is that these, first, are easier to understand, and second, capture the same concepts as continuous random variables.

Since pricing is always done from a relative perspective—that is, seeking to determine the asset price for a given reference date, around which all other relevant events are ordered and measured—we also model time in a relative way. Out of a series of dates $\mathbf{t} = \{-t_T, \ldots, -t_2, -t_1, 0, t_1, t_2, \ldots, t_T\}$ we denote by $t = 0$ as the reference or pricing date (we might be tempted to call this date "today"). This logic is illustrated in Figure 10.1.

In line with market conventions, we measure difference in time as the **year fraction** between two dates. Calendars have the unpleasant feature to be rather inhomogeneous (some months are 28, 29, 30, or 31 days in length), so the term year fraction is not a

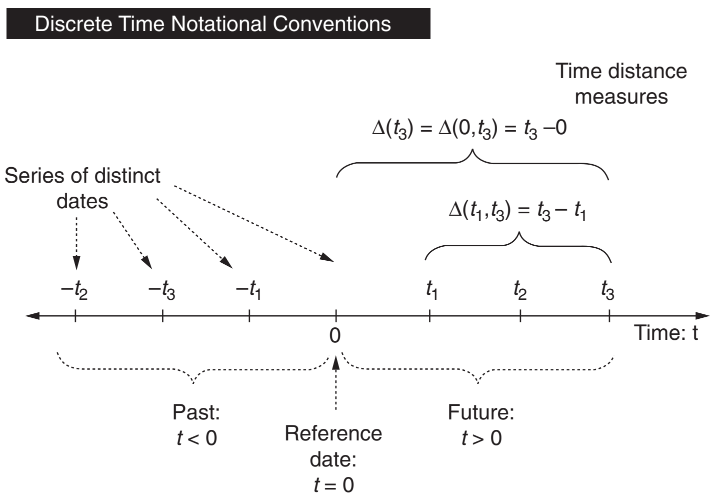


Figure 10.1: Discrete time notational conventions.

standardised concept. Rather, we must resort to a so-called **day count convention** that specifies the exact method to transform two dates into a year's fraction. There exist quite a number of day count conventions that are in use in the markets.

The most straightforward one is the rule ACT/ACT, which counts the number of actual calendar days between two dates and divides it by the actual number of calendar days in the current year. A related method is ACT/365, which always divides the first term of ACT/ACT by 365, even if the current year has 366 days. As these methods could reach widespread usage only with the emergence of computers, quite a number of simpler methods have been designed and are still in use.

In this book, however, we abstract from specific day count conventions by denoting the year fraction between two dates t_1 and t_2 as $\Delta(t_1,t_2)$, whatever may be the day count convention to be used. In addition, with reference to the central date $t = 0$ we specify $\Delta(t) = \Delta(0,t)$ making the formulation of certain formulas less heavy without any loss in precision.

10.2.4 Discount Factor and Future Value

Future values do not have the same relevance today that current values do. Think of a cake that you can eat tomorrow, but not today. Although you will still appreciate the outlook (if you like cakes), today's utility will be diminished by the fact that you have to wait another day to eat the cake.

Similarly, you can say that about money (cash flow) you will earn tomorrow, but not today. In the context of investing, another aspect becomes relevant: Money you have today can be invested and will bring you additional income, whereas money you have available only in the future cannot obviously be invested today. We call this concept the opportunity cost of time, the cost you incur due to the fact that the money is only available in the future is equal to the lost income. . . .

Hence, we may say in general that from today's perspective, future values are worth less today than they are tomorrow. The depreciation of future values to have their current day equivalent is captured by the concept of a **discount factor**.

We denote the discount factor that transforms a future cash flow x_t, $t > 0$ to get its value equivalent at date $t = 0$ as $m_{0,t}\ \forall t$. By construction $m_{0,t} < 1$. As there is no depreciation for current values we have $m_{0,0} = 1$.

However, there is no general rule what the "right" discount factor should be. As you will see next, determining the discount factor is one of the key issues when pricing an asset. To make things worse, the uncertainty inherent in future cash flows often makes even the discount factor uncertain from the perspective of the investor and, hence, a random variable.

10.3 The Mother of All Valuation Formulas

At its very heart, valuation theory was developed to operationalise the fundamental statement "the price of an instrument should equal the present value of its expected future cash flows it generates."

We start with developing an intuition of this by asking ourselves what the relevant elements of this statement are. The term "equal to" tells us that we have to formulate the problem in the form of an equation. On the left-hand side of the equation, there is the price of the instrument, call it p_t, the determinants of which we try to model. As we take the perspective of date $t = 0$, we may write the equation as $p_0 = ?$.

Things are more difficult on the right-hand side of the equation. What is meant by "present value of expected future cash flows"? Assume for the moment[2] that the instrument generates only one future cash flow at date t, $t > 0$.

- From the perspective of $t = 0$ the future cash flow is unknown, so we need to treat it as a random variable $\tilde{x}_t$ with domain **X**. Here, some may object that there are in fact assets where the (series of) future cash flow(s) is agreed upon at contract conclusion and, hence, is not really unknown. Classical examples are zero or straight coupon bonds. This is correct, but in particular from a risk management perspective, we should never forget that these future cash flows are "only" contractual agreements, and even if they are guaranteed by a third party to the contract, we can never be 100% certain that you will actually get the money in the exact amounts agreed upon. Even with the safest investment there does always exist a tiny probability of default, or nonpayment, or—at least—a reduction in amounts being paid. So, it is fair to treat the future cash flow as random and our analytical framework holds.
- In many cases, there is no explicit cash flow agreed between the investor in an asset and the issuer. For example in the case of equity shares or fund shares the investor has no guarantee that he or she will receive any income on that asset. However, the investor always has the possibility to resell the asset, making the future cash flow the unknown price he or she may expect to get.
- Having found a formulation for future cash flow, we can now turn to the statement "present value of …". It implies that from the perspective of $t = 0$, not the cash flow $\tilde{x}_t$ itself, but some transformation of it, call it $m_{0,t}(\tilde{x}_t)$, is to equal the current price. We will discuss this transformation at length later in this chapter. For now, it may be enough to know that this transformation is linear; that is, $m_{0,t}(\tilde{x}_t) = m_{0,t}\tilde{x}_t$, and that

[2] We will see later that once the single cash flow problem is solved properly, the extension to a multicash flow situation is a straightforward exercise, so this assumption does not limit the validity of our considerations.

the transforming factor is called **discount factor**. It has the subscript "0,*t*" as it performs a transformation from date t to date 0. In addition, at date $t = 0$ we do not know the exact value of the discount factor, as—we will see that later—it depends on the consumption level at date t, which itself is a function of $\tilde{x}_t$. Hence, we must also treat $m_{0,1}$ as a random variable, leading to the **stochastic discount factor** $\tilde{m}_{0,t} = f(c_t(\tilde{x}_t))$.

- The last missing element on the right-hand side is a formulation of the term "expected." As both, the discount factor $\tilde{m}_{0,t}$ and the ultimate cash flow $\tilde{x}_t$ are not known at date $t = 0$ we must build an expectation of their values using the information available at date $t = 0$, to be able to determine the current price, p_0. We express this by applying the expectation generating function $E_0[]$.

With these considerations, the initial statement may be expressed as

$$p_0 = E_0\left[\tilde{m}_{0,t}\tilde{x}_t\right] \tag{10-1}$$

where:

p_0	price of the asset at the reference date $t = 0$
$\tilde{x}_t$	unknown future cash flow due on date $t > 0$
$\tilde{m}_{0,t}$	stochastic (random) discount factor transforming the future cash flow $\tilde{x}_t$ into a value at date $t = 0$; that is, its "present value"
$E_0[]$	expectation generating function for the two random variables $\tilde{m}_{0,t}$ and $\tilde{x}_t$, using the information available at date $t = 0$

Equation (10-1) is called the **Central Pricing Equation**. It sets the scene for an important amount of theoretical and empirical research about investor behaviour and asset price setting in financial markets.[3] Most of valuation theory consists of specializations and manipulations of this formula. Despite its apparent simplicity, it is a very powerful tool that logically derives a fundamental relationship valid for any instrument price observed.

Figure 10.2 shows the various areas of valuation theory, all centred around the Central Pricing Equation. We approach this by presenting, step by step, the key areas in financial theory that bring subsequently a clearer picture on what is behind each of the elements in the Central Pricing Equation:

---

[3] For a detailed exposition of asset pricing theory, see Cochrane (2005), Duffie (2001), Huang and Litzenberger (1988), Ingersoll (1987).

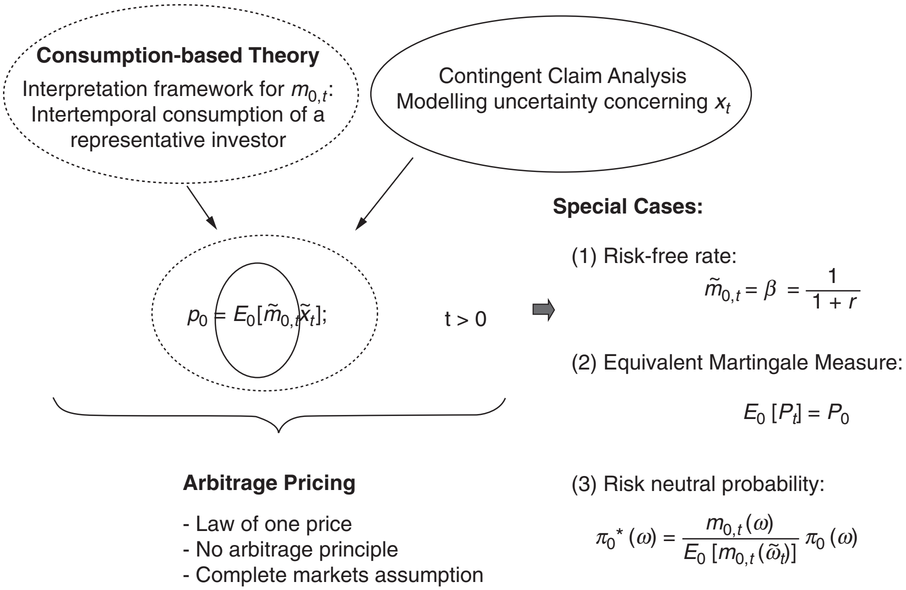


Figure 10.2: Nested theories explaining the Central Pricing Equation.

- **Consumption-based theory** models the investor's preferences and derives a structural discount factor formulation.
- **Contingent claim analysis** explicitly models a set of possible future cash flows. Discount factor is interpreted as the relative subjective value given by the investor to a particular state of nature.
- **Arbitrage pricing theory** analyses the question of whether and under which conditions an equilibrium market price exists and how it is connected to the central pricing equation.

For the reasons mentioned in Section 10.1, we will not explore the intellectually very interesting area of arbitrage pricing theory. The valuation approach we present here is fundamentally investor-oriented: For a given illiquid asset, the investor needs to make up his or her mind what an instrument is worth. By doing so he or she should already duly take repayment risks and opportunity costs into account. The question of whether this value can also be seen as a hypothetical market price is, in our opinion, not the most important at this stage.

10.4 Consumption-based Theory

We start with developing an explicit formulation of the stochastic discount factor $\tilde{m}_{0,t}$ based on a relatively simple model of investor behaviour. Consumption-based theory takes the point of view of an investor, asking for what effect an investment at date $t = 0$ could have on his or her current and expected future levels of consumption. The investment is assumed to be made by purchasing a financial instrument that is going to produce an (from the $t = 0$ perspective) uncertain future cash flow $\tilde{x}_t$, $t > 0$. At date $t = 0$, the investor must decide how much to save now ($t = 0$) and invest in a number n of instruments, each producing the uncertain cash flow $\tilde{x}_t$, while reducing the present day consumption.

We look at this problem in a two-date ($t = 0$; $t > 0$) context. This setting is quite realistic for many investment situations[4] since we do not specify the exact length of the interval between the two dates.

Suppose that the investor has a fixed noninvestment income over time that guarantees a time-independent fixed consumption level at all dates. The investor's default position would then be not to buy the instrument and to have access to a fixed (monetary) level of consumption, c^*, implied by noninvestment income.

At date $t = 0$, the investor's consumption level in monetary terms is therefore

$$c_0 = c^* - p_0 n \tag{10-2}$$

where p_0 is the price of the instrument and n is the number of instruments the investor decides to buy.

The future consumption level, c_t, $t > 0$, is not known, because it depends on the uncertain cash flow to be generated by the investment. So, from the perspective of $t = 0$, the future consumption level is again a random variable, $\tilde{c}_t$, which is determined by

$$\tilde{c}_t = c^* + \tilde{x}_t n \tag{10-3}$$

Here, $\tilde{x}_t$ is the cash flow generated by a single instrument, and n is the number of instruments purchased at date $t = 0$.

The two preceding equations described the budget constraints of the investor, who has to decide how many instruments (n) to buy. The decision criterion for the investor is his or her inter-temporal utility over present and future consumption levels. We write this inter-temporal utility as

[4] For example, this setting applies for any type of investment where there is only one expected future payoff, such as a zero coupon bond, a closed fund investment, an index certificate, or a floating rate note/bond, where only the next expected coupon payment is considered.

$$U(c_0, \tilde{c}_t) \tag{10-4}$$

Before entering the detailed decision problem in Section 10.4.3, we first have a look at the concept of utility and its operationalisation via a utility function in the next section.

10.4.1 Utility Functions and Investor Preferences

The investor needs to make sure that he or she draws the maximum utility, or pleasure, from his or her inter-temporal decision. Obviously, with unknown future values of wealth, and hence, consumption, this is a difficult one.

In formal language we use utility functions to summarize a consumer's behaviour by mapping the space of possible consumption bundles **C** to the real line **R**. The mapping is ordered in such a way that a higher value on the real line stands for a consumption bundle **c**, which is preferred by an individual to a consumption bundle **c′** with a lower value on the real line. Formally, this is

$$u : \mathbf{C} \rightarrow \mathbf{R} \text{ such that } \mathbf{c} \succ \mathbf{c}' \text{ if and only if } \boldsymbol{u}(\mathrm{c}) > \boldsymbol{u}(\mathrm{c}') \quad \forall \mathbf{c}, \mathbf{c}' \in \mathbf{C}$$

Working with actual consumption bundles (i.e., with bundles of goods) that may be consumed is realistic, but very unhandy in practice. We therefore prefer to refer to monetary consumption levels; that is, the monetary cost of goods being consumed.

It can be shown that if the preference ordering is complete, reflexive, transitive, and continuous—in short, representing a rationally behaving individual—it can be represented by a **continuous utility function**.

Figure 10.3 shows the classical form of a stylised continuous utility function over monetary consumption levels c, $u(c)$. It has the property that its first derivative is positive and the second derivative negative:

- $\partial u(c)/\partial c = u'(c) > 0 \quad \forall c$: The marginal utility of any consumption level is always positive, reflecting a desire for more consumption: People always want to consume more. The marginal utility is the infinitesimal (very small) change in utility due to a change in consumption.
- $\partial^2 u(c)/\partial c^2 = u''(c) < 0$: The marginal increase in utility is decreasing or constant, reflecting the declining marginal value of additional consumption. The last bite is never as satisfying as the first.

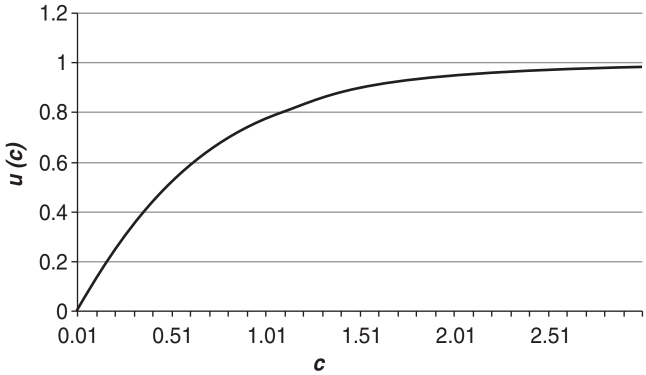


Figure 10.3: Concave utility function, $u(c)$, over monetary consumption levels c.

Functions with this property are called **concave**, hence we speak of a **concave utility function**.[5] An important property of utility functions is that any monotonic transformation of such a function represents the same preferences. Formally, if $u(\mathbf{c})$ represents some preferences $\mathbf{c} >= \mathbf{c}'$ and $f : R \to R$ is a monotonic function (i.e., has no breaks or steps in it), then $f(u(\mathbf{c}))$ will represent exactly the same preferences since $f(u(\mathbf{c})) \geq f(u(\mathbf{c}'))$ if and only if $u(\mathbf{c}) \geq u(\mathbf{c}')$.

On the Use of Monetary Amounts Instead of Real Quantities

From a purely theoretical perspective, we may argue that the utility expressed as a function of consumption rather depends on the quantity of physical goods than their monetary value. This would have the advantage that a unit of a consumption good can be used as a common denominator valid at all dates.

In practice, however, working with physical good representations brings along a rat-tail of significant data availability and measurement problems.

In addition, although the usage of physical goods may be of benefit for theoretical work on investor behaviour, the testing of these models usually is done using monetary measures, as they are more readily available and directly comparable.

It often is assumed in the literature therefore that the inflation rate is negligible with respect to the movements of asset prices (see, e.g., Cochrane 2001, Chapter 2). The introduction of the consumption plans for a physical good into the optimization objective indicates that this approach is not designed for direct use by fund managers. Instead, it is more suitable for individual behaviour analysis, for building macro dynamic models that link the real and financial sectors under the assumption of a representative agent, or else for asset pricing.

[5] See Chiang and Wainwright (2005) for an in-depth and easy-to-understand discussion of the algebra of utility functions.

In the context of uncertainty, when the utility over uncertain future consumption levels is measured, it can be shown that the expected utility over various possible monetary consumption levels must be in additive form. Consider the following situations:

1. In the first situation, you consume c^1 with probability π and c^2 with the (remaining) probability $1 - \pi$: $(\pi \rightarrow c^1, (1 - \pi) \rightarrow c^2)$.
2. In the second situation, you consume c^1 with probability θ and c^2 with the (remaining) probability $1 - \theta$: $(\theta \rightarrow c^1, (1 - \theta) \rightarrow c^2)$.

It can be shown[6] that there exists a continuous utility function u that describes the consumer's preferences; that is $(\pi \rightarrow c^1, (1 - \pi) \rightarrow c^2) \succ (\theta \rightarrow c^1, (1 - \theta) \rightarrow c^2)$ if and only if $u(\pi \rightarrow c^1, (1 - \pi) \rightarrow c^2) > (\theta \rightarrow c^1, (1 - \theta) \rightarrow c^2)$

This utility function $u(\ )$ is not unique; any monotonic transform would do as well.

Under some additional hypotheses, we can find a monotonic transformation of the utility function that has the very convenient **expected utility property**:

$$u\left(\pi \rightarrow x^1, (1-\pi) \rightarrow x^2\right) = \pi u\left(x^1\right) + (1-\pi) u\left(x^2\right)$$

The expected utility property says that the utility of a lottery is the expectation of the utility from its prizes. We can compute the utility of any lottery by taking the utility that would result from each outcome, multiplying that utility times the probability of occurrence of that outcome, and then summing over the outcomes. *Utility is additively separable over the outcomes and linear in the probabilities*.

Interpretation of Utility Functions

As you might have guessed from the many assumptions already needed, utility functions should not be over interpreted and mostly provide a convenient way to find an ordinal sorting of options and hence derive individual behaviour.

In particular they should not be given any psychological interpretation. The only relevant feature of a utility function is its ordinal character.

---

[6] See Varian (2006) for an intuitive explanation of the surrounding axioms, or—more advanced but having a benchmark character—Mas-Colell et al. (1995).

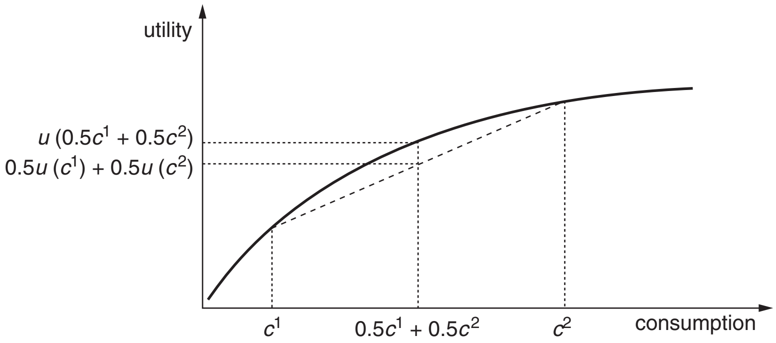


Figure 10.4: Expected utility of a lottery. The expected utility of the lottery is $0.5u(c^1) + 0.5u(c^2)$. The utility of the expected value of the gamble is $u(0.5c^1 + 0.5c^2)$. In this case, the utility of the expected value is higher than the expected utility of the lottery, so the individual is risk averse.

10.4.2 Risk Aversion and Risk Neutrality

Utility functions can be used to represent the attitude of investors toward risk. The expected utility property applies in particular in the case that uncertainty is with respect to different amounts of money; that is, cash flows an individual may receive under different circumstances ("states of nature").

It is hence possible to describe an individual's behaviour who is faced with uncertainty concerning future monetary outcomes, if we only know this particular representation of his or her utility function for monetary consumption levels. For example, to compute the consumer's expected utility in the situation $(\pi \rightarrow c^1, (1 - \pi) \rightarrow c^2)$, we just need to look at the corresponding expected utility $\pi u(c^1) + (1 - \pi)u(c^2)$.

This construction is illustrated in Figure 10.4 with $\pi = 0.5$. Notice that in this example the individual prefers to get the expected value rather than being exposed to the uncertain situation ("lottery"). That means that for the investor, the utility of the lottery $u(\pi \rightarrow c^1, (1 - \pi) \rightarrow c^2)$ is less than the utility of the expected value of the lottery, $\pi u(c^1) + (1 - \pi)u(c^2)$. Such a behaviour is called **risk aversion**.[7]

If an individual consumer is risk averse, the line drawn between any two points of the graph of his or her utility function must lie below the function. This is equivalent to the mathematical definition of a **concave function**. It implies that an agent prefers averages to extremes; this is equivalent to risk aversion. Hence, concavity of the expected utility function is equivalent to risk aversion.

[7] An individual may also be **risk-loving**; in such a case, the consumer prefers a lottery to its expected value.

It is often convenient to have a measure of risk aversion. Intuitively, the more concave the expected utility function, the more risk averse the consumer. Thus, we might think we could measure risk aversion by the second derivative of the expected utility function that measures the curvature, or concavity of the function. However, this definition is not invariant to changes in the expected utility function: if we multiply the expected utility function by 2, the consumer's behaviour doesn't change, but our proposed measure of risk aversion does.[8] However, if we normalize the second derivative by dividing by the first, we get a reasonable measure, known as the **Arrow-Pratt measure of (absolute) risk aversion**:

$$AP(w) = \frac{u''(w)}{u'(w)}$$

An important special case is the **risk-neutral individual**: Risk-neutrality implies that the consumer is indifferent between a sure expected value and the expected utility of it. Formally, this is expressed by a linearly increasing utility function:

$$u'(c) = k \quad \forall c$$

$$u''(c) = 0 \quad \forall c$$

The implication of this situation is shown in Figure 10.5. A risk neutral individual is indifferent on whether he or she gets the expected value of a lottery or the lottery itself.

10.4.3 The Investor's Decision

With the utility function toolkit at hand, we can now formulate the investor's decision problem at date $t = 0$.

Using the formulations for c_0 in Equation (10-4) and $\tilde{c}_t$ in Equation (10-5), we model the investor by the **expected utility function** defined over current and future values of consumption:

$$U_0 = E_0\left[U\left(c_0, \tilde{c}_t\right)\right] = u\left(c_0\right) + \beta E_0\left[u\left(\tilde{c}_t\right)\right] \qquad (10\text{-}5)$$

From Equation (10-4) we know that the future consumption $\tilde{c}_t$ is random as it depends on the future cash flow $\tilde{x}_t$. Discounting the future by β, $0 \le \beta \le 1$, captures impatience, and β is called the **subjective discount factor**. Note that we use the expectation generator $E_0[]$ to

8 $v(u(c)) = 2u(c) \rightarrow \partial^2 v(\ )/\partial c^2 = 4u''(c)$, which is larger than $\partial^2 u(\ )/\partial c^2 = u''(c)$.

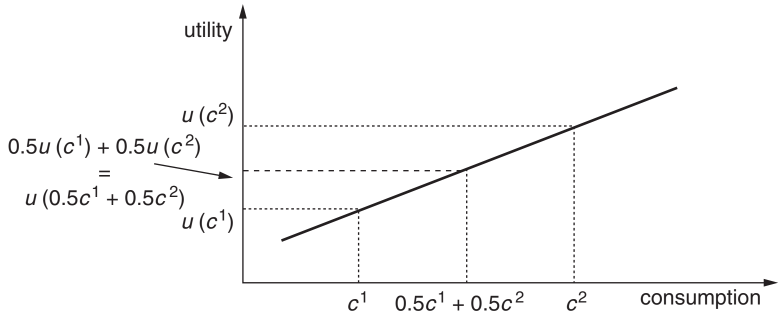


Figure 10.5: Risk neutrality.

account for uncertain states and their probabilities. In Section 10.5 we will make this formulation more explicit.

Because we do not want to enter a structural discussion of the Central Pricing Equation,[9] it will suffice to make the standard concavity assumptions about the utility function; that is, the direct utility function $u(\ )$ is increasing, reflecting a desire for more consumption, and concave, reflecting the declining marginal value of additional consumption units.

In line with Section 10.4.2, this formalism captures our investor's impatience and his or her aversion to risk. Both assumptions are enough to derive an optimum characterisation of the investor's inter-temporal decision.

Our objective is to find the value at time $t = 0$ of the single future cash flow $\tilde{x}_t$, $t > 0$. This future cash flow structure consists of the future price of the instrument at date t, $\tilde{p}_t$ and (possibly) an additional income flow $\tilde{d}_t$; that is, $\tilde{x}_t = \tilde{p}_t + \tilde{d}_t$.

Assuming that the investor can freely buy or sell as much of the payoff $\tilde{x}_t$ as he or she wishes, how much will he or she buy or sell? The problem is to find the optimal number of assets n^* that maximises the inter-temporal utility, Equation (10-6), under the constraint that this investment will decrease the current consumption level in $t = 0$, c_0. Formally this is

$$\max_n U(c_0, \tilde{c}_t) = u(c_0) + \beta E_0[u(\tilde{c}_t)] \quad s.t.$$

$$c_0 = c^* - p_0 n$$

$$\tilde{c}_t = c^* + \tilde{x}_t n$$

9 There exists a large branch of financial economics that tries to explicitly specify a functional form for a "typical" utility function. Besides the fact that such an honourable endeavour is of little relevance for our purposes, they usually are confronted with the full arsenal of practical complications, like validity of representativeness assumption, non-availability of appropriate micro-data, just to name a few. What is important for us is that you take the theory as a tool to structure your thinking about prices of (financial) assets, but nothing more.

Substituting the constraints into the objective function, and setting the derivative with respect to n equal to zero,[10] we obtain the first-order condition for an optimal consumption and portfolio choice,

$$p_0 u'(c_0) = E_0\left[\beta_{0,t} u'(\tilde{c}_t)\,\tilde{x}_t\right] \qquad (10\text{-}6)$$

The investor continues to buy or sell the instrument until the marginal loss equals the marginal gain. The investor buys more or less of the instrument until this first-order condition holds. Equation (10-6) expresses the condition for an optimum:

- $p_0\, u'(c_0)$ is the marginal loss in utility if the investor buys another unit of the instrument
- $E_0[\beta_{0,t} u'(\tilde{c}_t)\,\tilde{x}_t]$ is the marginal expected increase in (discounted) utility the investor obtains from the extra payoff at t

The marginal utility loss of consuming a little less today and buying a little more of the instrument should equal the marginal utility gain of consuming a little more of the asset's payoff in the future.

If the price and payoff of the asset do not satisfy this relation, the investor will adapt the amount of the asset he or she holds (i.e., buy more or less of the asset) and, hence, implicitly adjust the amount of consumption.

As c_0 is known, we can say that $u'(c_0) = E_0[u'(c_0)]$ and therefore write

[10] The optimality condition implies the first derivative of $U(\ )$ with respect to n being equal to 0. Inserting the restrictions (10-3) and (10-4) into $U(\ )$ yields:

$$U(c_0, \tilde{c}_t) = u(c^* - p_0 n) + \beta E_0[u(c^* + \tilde{x}_t n)]$$

Taking the first derivative with respect to n and equalling it to zero produces

$$\frac{\partial U(c_0, \tilde{c}_t)}{\partial n} = u''(c_0)(-p_0) + \beta E_0[u'(\tilde{c}_t)\,\tilde{x}_t] \stackrel{!}{=} 0$$

The second condition for a maximum is that the second derivative of the objective function with respect to n is negative over all values of c, i.e.,

$$\frac{\partial U(c_0, \tilde{c}_t)}{\partial n} = u''(c_0)(-p)^2 + \beta E_0\left[u''(\tilde{c}_t)(\tilde{x}_t)^2\right] < 0$$

As $\partial^2 u(\ )/\partial c^2 = u''(\ )$, this is by definition smaller than zero.

$$p_0 = E_0\left[\beta_{0,t}\frac{u'(\tilde{c}_t)}{u'(c_0)}\tilde{x}_t\right] \quad (10\text{-}7)$$

It follows that the asset's price should equal the expected discounted value of the asset's payoff, using the investor's marginal utility to discount the payoff.

You may have noticed that Equation (10-7) is actually very close to the Central Pricing Equation. Indeed, if we define the **stochastic discount factor**

$$\tilde{m}_{0,t} \equiv \beta\frac{u'(\tilde{c}_t)}{u'(c_0)} \quad (10\text{-}8)$$

Then, we again get the basic pricing formula, Equation (10-7):

$$p_0 = E_0\left[\tilde{m}_{0,t}\tilde{x}_t\right] \quad (10\text{-}9)$$

From Equation (10-9) we see that the discount factor is a function of a random variable $\tilde{c}_t$, is a random variable itself, and hence follows a distribution. From Equation (10-4) we know that $\tilde{c}_t$ is a function of $\tilde{x}_t$ and so it is $\tilde{m}_{0,t}$. This means that $\tilde{m}_{0,t}$ and $\tilde{x}_t$ are correlated. $\tilde{m}_{0,t}$ is hence called **stochastic** (another word for random) because it is not known with certainty at time $t = 0$. Because of Equation (10-8), $\tilde{m}_{0,t}$ also often is called the **marginal rate of substitution**, the rate at which the investor is willing to substitute consumption at time t for consumption at time $t = 0$. net.

Example: The Marginal Rate of Substitution (MRS)

Let, $u(\mathbf{c}) = u(c_0, c_t)$ be an inter-temporal utility function of present ($t = 0$) and future ($t > 0$) levels of consumption. Suppose that we increase the future level of consumption c_t; how does the consumer have to change his or her consumption at date $t = 0$ in order to keep the inter-temporal utility constant?

Let dc_0 and dc_t be the changes in c_0 and c_t. By assumption, the change in utility must be zero, so

$$\frac{\partial u(\mathbf{c})}{\partial c_0} dc_0 + \frac{\partial u(\mathbf{c})}{\partial c_t} dc_t = 0$$

Hence,

$$\frac{dc_0}{dc_t} = -\frac{\frac{\partial u(\mathbf{c})}{\partial c_t}}{\frac{\partial u(\mathbf{c})}{\partial c_0}}$$

This expression is known as the **marginal rate of substitution** between consumption levels c_0 and c_t. The marginal rate of substitution does not depend on the utility function chosen to represent the underlying preferences.[11]

10.4.4 Multiple Periods

So far, we have analysed the investor's decision in a two-date (or two-period) context. We pointed out that this setting captures a wide range of practical situations, in particular if we interpret $\tilde{x}_t = \tilde{p}_t + \tilde{d}_t$. Using this, we can now formally show that the central pricing equation holds for any two periods of a multi-period model.

Think of the future cash flow $\tilde{x}_t$, consisting of the future price $\tilde{p}_t$ and, possibly, an income cash flow $\tilde{d}_t \left(\text{i.e., } \tilde{x}_t = \tilde{p}_t + \tilde{d}_t\right)$, all random variables, unknown at date $t = 0$. This formulation allows us to think of multi-period pricing problems in just two-period contexts.

In situations, however, where a set of future cash flows is prescheduled and hence known at date $t = 0$, we will want to relate a price to an entire cash flow stream, rather than just to

[11] To see this, let $v(u)$ be a monotonic transformation of utility. The marginal rate of substitution for this utility function is

$$\frac{dx_j}{dx_i} = -\frac{v'(u)}{v'(u)} \frac{\frac{\partial u(\mathbf{x})}{\partial x_i}}{\frac{\partial u(\mathbf{x})}{\partial x_j}} = -\frac{\frac{\partial u(\mathbf{x})}{\partial x_i}}{\frac{\partial u(\mathbf{x})}{\partial x_j}}$$

The fraction on the left is the marginal rate of substitution between good i and j, and the fraction on the right might be called the economic rate of substitution between goods i and j. Maximization implies that those two rates of substitution be equal.

one dividend and next period's price.[12] Using the consumption-based approach, we extend Equation (10-5) to a multi-period utility function,

$$U_0 = E_0\left[\sum_{t=0}^{T} \beta^t u(c_t)\right]$$

Note that, as $\beta^0 = 1$, for $T = 1$, this is exactly the two-period utility function of Equation (10-4). As with the two-period model, the investor's first-order condition gives the pricing formula:

$$p_0 = E_0\left[\sum_{t=0}^{T} \beta^t \frac{u'(\tilde{c}_t)}{u'(c_0)} \tilde{x}_t\right] = E_0\left[\sum_{t=0}^{T} \tilde{m}_{0,t}\tilde{x}_t\right] \tag{10-10}$$

which for $T = 1$ is equivalent to the two-period Central Pricing Equation. An interesting formulation of Equation (10-11) is obtained when using $\tilde{x}_t = \tilde{d}_t + \tilde{p}_t$ incrementally to derive a series of future cash flows.

$$p_0 = E_0\left[\tilde{m}_{0,t}\tilde{x}_t\right] = E_0\left[\tilde{m}_{0,t}\left(\tilde{d}_t + \tilde{p}_t\right)\right] = E_0\left[\tilde{m}_{0,t}\tilde{d}\right] + E_0\left[\tilde{m}_{0,t}\tilde{p}_t\right]$$

As

$$p_t = E_t\left[\tilde{m}_{t,t+1}\left(\tilde{d}_{t+1} + \tilde{p}_{t+1}\right)\right]$$

We can write p_0 as

[12] An example for such a cash flow stream might be a fixed coupon bond, where we have a number of income (coupon) payments and, at date T, the redemption price, plus possibly a final income payment:

$$\mathbf{x} = [x_1, x_2, \ldots, x_{1T}] = [d_1, d_2, \ldots, d_T + p_i]$$

Note that periods between the dates do not need to be equidistant for the approach to hold.

$$
\begin{aligned}
p_0 &= E_0\left[\tilde{m}_{0,t}\tilde{d}\right] + E_0\left[\tilde{m}_{0,t}E_t\left[\tilde{m}_{t,t+1}\left(\tilde{d}_{t+1} + \tilde{p}_{t+1}\right)\right]\right] \\
&= E_0\left[\tilde{m}_{0,t}\tilde{d}\right] + E_0\left[\tilde{m}_{0,t}\tilde{m}_{t,t+1}\tilde{d}_{t+1}\right] + E_0\left[\tilde{m}_{0,t}\tilde{m}_{t,t+1}\tilde{p}_{t+1}\right]
\end{aligned} \tag{10-11}
$$

Acknowledging $\tilde{m}_{0,t}\tilde{m}_{t,t+1} = \tilde{m}_{0,t+1}$ and extending the logic for $\tilde{p}_{t+1}$, $\tilde{p}_{t+2}$, … etc., this is equivalent to the formulation in Equation (10-10).

10.5 Contingent Claim Analysis

The consumption based theory introduces a framework for the interpretation of the stochastic discount factor $\tilde{m}_{0,t}$ as the marginal rate of substitution between current and future consumption. It says little, however, about the uncertainty inherent to the future cash flow $\tilde{x}_t$ and how to deal with this uncertainty. This is the contribution of contingent claim analysis, which explicitly models and analyses the effect of uncertainty and the corresponding expectation building. Historically, contingent claims analysis allowed extending the relationship between competitive equilibriums and welfare optimality to an economy operating under uncertainty. Arrow (1953, reprinted 1964) introduced uncertainty into a pure-exchange economy via a random variable designating the "state of nature," $\tilde{\omega}$. Debreu (1959) extended Arrow's pure exchange model in important ways, hence the name "Arrow-Debreu" to describe the contingent claims economies and related concepts.[13]

10.5.1 States of Nature and Contingent Claims

We start with formalising the concept of an *ex ante* unknown "state of nature at date *t*." By *ex ante* we mean from the point of view of any date prior to *t*.

The "state of nature at date at date *t*" concept satisfies the definition for a random variable in Section 2.2. We denote this random variable as $\tilde{\omega}_t$ with a domain set $\mathbf{\Omega} = \{\omega^1, \omega^2, \ldots, \omega^\Omega\}$ containing Ω mutually exclusive **states of nature** that may occur at the future date *t*. A particular realisation of $\tilde{\omega}_t$ is ω_t, being one of the elements in $\mathbf{\Omega}$; that is, $\omega_t \in \mathbf{\Omega}$. The exact value ω_t is not revealed until date *t*, hence the time-specific subscript.

In its earliest formulations, contingent claim analysis considered states representing physical conditions of the environment; for example, "rain" or "shine." But the notion of a state soon became and can be more broadly interpreted as representing other forms of uncertainty in financial markets, such as "issuer default," "underlying price above/below strike price," and so on.

[13] Both Arrow and Debreu are later awarded the Nobel Prize in Economics for their contribution in this very field.

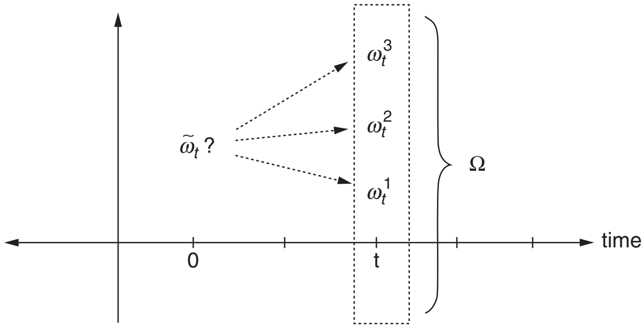


Figure 10.6: Two-period (date) example of the random variable "state of nature at date *t*," $\tilde{\omega}_t$. At date *t* = 0, its value is unknown, but one of three possible states, either ω_t^1, ω_t^2, or ω_t^3, is revealed at date *t*.

The set-up is depicted in Figure 10-6 where, at date $t = 0$, the future state is unknown and represented by the random variable $\tilde{\omega}_t$. At a date $t > 0$ a specific value from the domain Ω emerges.

Given $\tilde{\omega}_t$, a **contingent claim** is defined as a contractual claim on a unit monetary cash flow (e.g., one Euro or one Great Britain Pound). These claims are traded and agreed upon at date $t = 0$, prior to the realization of state $\tilde{\omega}_t$. It can be shown[14] that this trading results in a competitive equilibrium and hence the resulting outcome is stable. At date *t*, trading stops, the state of nature, ω_t, is revealed, and only those contingent claims that are conditional on the realized state, $\tilde{\omega}_t = \omega_t$, are paid out.

In the case where a certain state of nature ω_t materialises at date $t > 0$, the corresponding contingent claim pays one monetary unit, and zero otherwise. Formally, this is expressed as

$$\text{Contingent Claim} = 1\left[\tilde{\omega}_t = \omega_t\right] = \begin{cases} 1 & \text{if } \tilde{\omega}_t = \omega_t \\ 0 & \text{if } \tilde{\omega}_t \neq \omega_t \end{cases} \qquad \left| \omega_t \in \Omega \right.$$

Due to their contractual nature, contingent claims may be interpreted as **elementary securities**. Elementary securities generate a cash flow only in one single state of nature. Contingent claims are elementary securities with the special feature that their cash flow is a unitary value. We will use this important property in the next section.

[14] See Arrow (1964); more developed, Nagatani (1975); and for a summary, Radner (1982).

In his 1953 article, Arrow assumed that there exist precisely as many contingent claims as there are possible states of nature, with states being mutually exclusive[15] so that only one of all possible states can materialise at date t.

In this setting, the Ω-dimensional payoff vectors of contingent claims, each claim paying off 1 monetary unit if $\tilde{\omega}_t = \omega_t$ and zero in all other states, are linearly independent. The matrix of claim-specific unitary cash flows may be represented by a Ω-dimensional identity matrix

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & & \\ \vdots & & \ddots & 0 \\ 0 & & 0 & 1 \end{bmatrix} \sim (\Omega \times \Omega) \qquad (10\text{-}12)$$

The contingent claims in this particular setting are called **Arrow-Debreu securities**, $1[\tilde{\omega}_t = \omega_t]$.

10.5.2 Contingent Claims and Cash Flows

Contingent Claims Analysis further distinguishes between the unit monetary contingent claims, $1[\tilde{\omega}_t = \omega_t]$, and the actual cash flows that are generated if a given state is adopted. Each state ω corresponds to a specific cash flow, $x_t(\omega)$, out of a set of possible cash flows $\mathbf{X}$. At date $t = 0$, the actual cash flow generated at $t = 0$ is unknown and hence modelled as a random variable $\tilde{x}_t$.

The various realisations of $\tilde{x}_t$ may be mapped to states of nature $\tilde{\omega}_t$. Denote this mapping as $\tilde{x}_t = x(\tilde{\omega}_t)$. The mapping from $\tilde{\omega}_t$ to $\tilde{x}_t$, $\tilde{\omega}_t \rightarrow \tilde{x}_t$, is non-reflexive, meaning that for each ω in $\mathbf{\Omega}$ one may exactly identify a specific x, but not the other way around. It is therefore possible that multiple states of nature result in the same payment! This is illustrated in Table 10.1, where two states, ω^1 and ω^3, result in the same cash flow, x^1. Formally, this may be expressed as

$$\tilde{\omega}_t \rightarrow \tilde{x}_t \Leftrightarrow \tilde{x}_t = x(\tilde{\omega}_t) = \begin{cases} x^1 & if \quad \tilde{\omega}_t = \{\omega^1, \omega^3\} \\ x^2 & if \quad \tilde{\omega}_t = \omega^2 \end{cases} \qquad (10\text{-}13)$$

[15] As shown in Section 10.2.2, the proper definition of possible states, in particular their mutual exclusiveness, is key to having a meaningful definition of a random variable.

Table 10.1: Exemplary mapping from $\tilde{\omega}_t$ to $\tilde{x}_t$ at date $t > 0$

State $\tilde{\omega}_t$ at date t		Cash flow $\tilde{x}_t$ at date t
ω^1	$\rightarrow$	x^1
ω^2	$\rightarrow$	x^2
ω^3	$\rightarrow$	x^1

or, more generic

$$\tilde{x}_t = \sum_{\omega_t \in \Omega} 1\left[\tilde{\omega}_t = \omega_t\right] x_t(\omega_t)$$

10.5.3 Contingent Claims and State Prices

From the point of view of the investor, the realisation of different states of nature does not have the same value, and hence the monetary unit resulting from the corresponding claims are not of equal **subjective value**. How is that?

The usefulness of a good often depends on the circumstances (or "state of nature") in which it becomes available: An umbrella when it is raining has a different appeal than an umbrella when it is not raining. Another example is the amount of ice cream that can be consumed in hot weather and cold weather. Although the actual ice cream will cost the same, the utility drawn from its consumption is most likely much higher in hot than in cold weather. A more serious example involves health insurance: How much would a million Euros be worth to someone lying in a coma?

As the investor is interested primarily in the utility drawn from consumption levels, it is important to distinguish levels of consumption by the state of nature in which they may be consumed.

This difference in utility is captured by the **state price** concept, $q_0(\omega_t)$, measuring the value the investor attributes to receiving a unit monetary value in a specific state ω_t. It is also the price at date $t = 0$ of a contingent claim (Arrow-Debreu security), and hence also is called **contingent claim price**.

With this concept in mind, we can decompose assets promising an uncertain future cash flow $\tilde{x}_t$ into Ω elementary securities, each consisting of $x(\omega_t)$ Arrow-Debreu securities

Table 10.2: States of nature and corresponding cash flows

State Description	State at date *t*	Cash flow at date *t*
Bad weather	ω^1	x^1 = 100 EUR
Good weather	ω^2	x^2 = 5 EUR

$1[\tilde{\omega}_t = \omega_t]$. Every such elementary security represents the cash flow received in a specific state and, as utility differs for each state, is separately priced. To see how this works, consider the example in Table 10.2.

In this example, there are two states of nature, $\Omega = \{\omega^1, \omega^2\}$, and two possible cash flow values $\mathbf{X}_t = \{x^1, x^2\} = \{100, 5\}$ that are contingent according the mapping set out in Table 10.2. As each Arrow-Debreu security yields exactly one Euro in the corresponding state of nature, the value for the investor of receiving the 100 Euros in bad weather is

$$q_0(\omega^1)100 = q_0(\omega^1)x^1,$$

and the value of receiving the five Euros in good weather is

$$q_0(\omega^2)5 = q_0(\omega^2)x^2$$

$x_t(\omega)$ denotes the asset's cash flow in state of nature $\tilde{\omega}_t = \omega$. Using $q_0(\omega)$ and $x_t(\omega)$ the decomposition of the asset into a bundle of orthogonal[16] contingent claims, each potentially resulting in a different cash flow, the asset's price must then equal the value of the contingent claims of which it is a bundle,

$$p_t^j(\mathbf{x}_t^j) = \sum_{\omega \in \Omega} q_0(\omega) x_t(\omega) \tag{10-14}$$

[16] Orthogonality is equivalent in this case to mutual exclusiveness and comes from the requirement that the states of nature in Ω are mutually exclusive, so if one state emerges no other state can emerge as well. If a claim is realized (i.e., a particular state of nature has emerged in date t), then all other contingent claims not corresponding to this state of nature are void as no two or more contingent claims can materialise at the same time.

10.5.4 Investor Decision under Uncertainty

How should we interpret the notion of a state price and the pricing formula, Equation (10-14), and how is this related to the Central Pricing Equation? Why is one state preferred over another and hence given a higher value by the investor? To find this out, we again take a look at the decision problem of an individual in the consumption-based context, this time explicitly modelling the uncertainty across various states of nature.

From the contingent claim framework we now have an explicit formulation of the future consumption level, $\tilde{c}_t = c(\tilde{\omega}_t)$, as a function of the state of nature. Start with transforming the budget constraint for the future consumption level by using the mapping from $\tilde{\omega}_t$ to $\tilde{x}_t$, $\tilde{x}_t = x(\tilde{\omega}_t)$.

$$\tilde{c}_t = c_t(\tilde{\omega}_t) = c^* + x_t(\tilde{\omega}_t)n \tag{10-15}$$

Next, we adjust the utility function to account for uncertainty in a more explicit way. Denote $\pi_0(\omega) = \Pr(\tilde{\omega}_t = \omega | \omega \in \mathbf{\Omega})$ as the probability as perceived by the investor at date t = 0 that state ω will emerge (i.e., $\tilde{\omega}_t = \omega$) at date t. The probabilities $\pi_0(\omega)$are the investors' **subjective probabilities** for the various states. Asset prices are set, after all, by investors' demands for assets, and those demands are set by investors' subjective evaluations of the probabilities of various events. Using Equation (10-15) we can formulate the expectation of the future utility as

$$E_0[u(\tilde{c}_t)] = \sum_{\omega \in \Omega} \pi_0(\omega) u(c_t(\omega))$$

The investor's decision problem may then be written as

$$\max_n E_0[U(c_0, c_t(\tilde{\omega}_t))] = u(c_0) + \beta \sum_{\omega \in \Omega} \pi_0(\omega) u(c_t(\omega)) \quad s.t.$$

under the budget constraints

$$c_0 = c^* - p_0 n$$
$$c_t(\omega) = c^* + x_t(\omega) n$$

Note that we have proper probabilities for the various state of the lottery and this expectation shows how Equation (10-6) satisfies the expected utility criterion needed for working with additive utility functions in the context of uncertainty.

As in Chapter 4, we insert the restrictions into the objective function to get

$$\max_n E_0\left[U\left(c_0, c_t(\tilde{\omega}_t)\right)\right] = u\left(c^* - p_0 n\right) + \beta \sum_{\omega \in \Omega} \pi_0(\omega) u\left(c^* + x_t(\omega) n\right)$$

The first-order condition for an optimum of this is:[17]

$$\frac{\partial E_0\left[U\left(c_0, c_t(\tilde{\omega}_t)\right)\right]}{\partial n} = -p_0 u'(c_0) + \beta \sum_{\omega \in \Omega} \pi_0(\omega) u\left(c_t(\omega)\right) x_t(\omega) \overset{!}{=} 0$$

This is analogous to Equation (10-6) in Section 10.4.3 and results in

$$p_0 = \sum_{\omega \in \Omega} \pi_0(\omega) \beta \frac{u'\left(c_t(\omega)\right)}{u'(c_0)} x_t(\omega) \tag{10-16}$$

which is nothing else than a more detailed formulation of the Central Pricing Equation $p_0 = E_0[\tilde{m}_{0,t}\tilde{x}_t]$ for the case that the uncertainty surrounding $\tilde{x}_t$ is described by the state of nature random variable $\tilde{\omega}_t$.

Further differentiating Equation (10-16) with respect to $x_t(\omega)$ provides—in analogy to Equation (10-14)—a structural expression linking the state price $q_0(\omega)$ to the marginal utility of the investor:

$$q_0(\omega) = \frac{\partial p_0}{\partial x_t(\omega)} = \pi_0(\omega) \beta \frac{u'\left(c_t(\omega)\right)}{u'(c_0)} = \pi_0(\omega) m_{0,t}(\omega) \tag{10-17}$$

[17] The second order condition is

$$\frac{\partial^2 E_0[U(c_0, c_t(\tilde{\omega}_t))]}{\partial n^2} = -p_0 u'(c_0) = \beta \sum_{\omega \in \Omega} \pi_0(\omega_t) u(c_t(\omega_t)) x_t(\omega_t) < 0$$

as $u''(\,) < 0$, this expression becomes < 0.

This illustrates how the state price measures the value of the emergence of a state (and resulting cash flow) for the investor in terms of utility. It also brings along the interpretation of the stochastic discount factor $\tilde{m}_{0,t}(\tilde{\omega}_t)$ as a set of contingent claim prices, scaled by probabilities. As a result of this interpretation, the combination of discount factor and probability is sometimes called **state-price density**:

$$m_{0,t}(\omega) = \frac{q_0(\omega)}{\pi_0(\omega)} = \beta \frac{u'(c_t(\omega))}{u'(c_0)}$$

This formulation also shows that the randomness in the discount factor comes from the uncertainty about the future state of nature.

Assuming smooth utility functions, the same logic as in Section 10.4 ensures that state prices $q_0(\omega_t)$ are proportional to each investor's marginal rate of substitution between current and future consumption, that is

$$q_0(\tilde{\omega}_t) \approx \frac{\partial U^i / \partial c_t(\tilde{\omega}_t)}{\partial U^i / \partial c_0} = \beta \frac{u'(c_t(\tilde{\omega}_t))}{u'(c_0)} =: MRS^i_{0,\omega}$$

The investor's first-order conditions say that the marginal rate of substitution between **states** at date t equals the relevant price ratio for the case that $\pi_0(\omega^1) = \pi_0(\omega^2)$,

$$\frac{m_{0,t}(\omega_t^1)}{m_{0,t}(\omega_t^2)} = \frac{u'(c(\omega_t^1))}{u'(c(\omega_t^2))}$$

$m(\omega^1)/m(\omega^2)$ gives the rate at which the investor can give up consumption in state 2 in return for consumption in state 1 through purchase and sales of contingent claims.

We learn that the discount factor m is the marginal rate of substitution between date—*and state*—contingent commodities. That is why it, like $c(\omega)$, is a random variable.

Figure 10.7 gives the economics behind this approach to asset pricing. The investor's first-order conditions in a contingent claims market implies that the marginal rate of substitution equals the discount factor and the contingent claim price ratio.

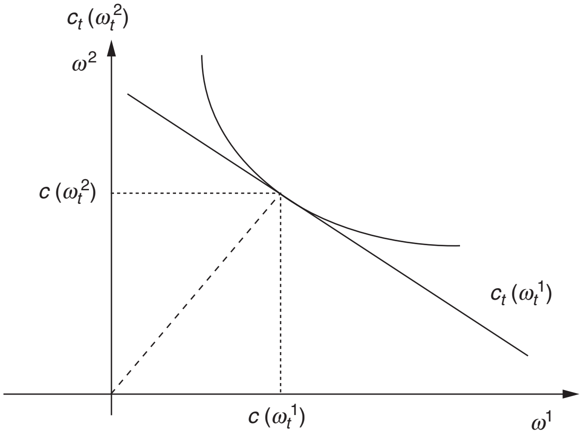


Figure 10.7: Indifference curve and contingent claim prices.

10.5.5 Special Case I: The Risk-free Rate

An important concept to which we will have much recourse is the **risk-free rate**, which we can now define and characterise.

Think of an instrument that generates the same cash flow whatever the state of nature; that is, where $x_t(\omega) = x_t^f \ \forall \omega \in \Omega$. We call this instrument the risk-free instrument and the return it generates, the risk-free rate.

The risk-free rate is the ratio of future cash flow to current price, x_t^f / p_0, for an instrument that guarantees (by all means of probability) the same cash flow in all possible states Ω; that is,

$$1 + r_{0,t}^f = \frac{x_t^f}{p_0} \text{ for } x_t(\omega) = x_t^f \quad \forall \omega \in \Omega$$

Accordingly, $r_{0,t}^f$ is called the **net risk-free rate** from date 0 to date t and $1 + r_{0,t}^f$ stands for the corresponding **gross risk-free rate**.

10.5.6 Special Case II: Equivalent Martingale Measures

Another important concept that is fundamental for derivative pricing is the **equivalent martingale measures**. A martingale is a stochastic process $\tilde{x}_t$ such that at any date $t = 0$ its conditional expectation for date $t > 0$ coincides with its current value: $E_0[\tilde{x}_0] = x_0$. In discrete time (i.e., when time is measured by separable dates), a martingale is a process the value of which at any date is equal to its conditional expectation for one date later.

Price processes, p_t, are in general not martingales, meaning that empirically they usually exhibit a drift or trend. If we recognise that prices are equal to the discounted values of expected future cash flows, it can be demonstrated that in the absence of arbitrage there is a probability measure in which all price processes, appropriately discounted, become martingales. The law of iterated expectation then implies that price plus payoff processes are martingales.

To illustrate this, assume a continuity of dates starting with $t = 0$, that is, $\mathbf{t} = \{0, 1, 2, 3, \ldots, T\}$. If we use the risk-free rate from date 0 to t, $r_{0,t}^f$, to discount future cash flows, we may formulate the Central Pricing Equation as

$$p_0 = E_0^Q\left[\sum_{t=1}^{T} \tilde{m}_{0,t}\tilde{x}_t\right] = E_0^Q\left[\sum_{t=1}^{T} \frac{1}{1+r_{0,t}^f}\tilde{x}_t\right]$$

Note that this formulation presupposes the absence of arbitrage, a market-price generating concept that implies that the correct discount factor for market prices is the risk-free rate. We use this concept repeatedly in this book.

It is also important to note that this approach deviates from the consumption-based theory as it takes away the investor-specific inter-temporal preferences in its quest for statistically convenient, "investor-neutral" valuation approach. This has important consequences, when interpreting valuations that are done using the equivalent martingale measure approach.

Acknowledge further that the series of cash flows generated by the instrument is

$$\mathbf{d} = \left[\tilde{d}_1, \tilde{d}_2, \ldots, \tilde{d}_T\right]$$

With the definition of the uncertain future cash flow at date t, $\tilde{x}_t = \tilde{p}_t + \tilde{d}_t$, we can write the price of the instrument being discounted by risk-free rates as

$$p_0 = E_0^Q\left[\sum_{t=1}^{T} \frac{1}{1+r_{0,t}^f}\tilde{x}_t\right] = E_0^Q\left[\frac{1}{1+r_{0,1}^f}\tilde{d}_1 + \sum_{t=2}^{T} \frac{1}{1+r_{0,t}^f}\tilde{d}_t\right]$$

From this formulation we know that

$$m_{0,2} = m_{0,1} \cdot m_{1,2}$$

which, using risk free rates, is equivalent to

$$\frac{1}{1+r_{0,2}^f} = \frac{1}{1+r_{0,1}^f} \cdot \frac{1}{1+r_{1,2}^f}$$

Hence we can write

$$p_0 = E_0^Q\left[\frac{1}{1+r_{0,1}^f}\tilde{d}_1 + \frac{1}{1+r_{0,1}^f} E_1^Q\left[\sum_{t=2}^{T}\frac{1}{1+r_{1,t}^f}\tilde{d}_t\right]\right]$$

which, as

$$p_1 = E_1^Q\left[\sum_{t=2}^{T}\frac{1}{1+r_{1,t}^f}\tilde{d}_t\right]$$

is equal to

$$p_0 = E_0^Q\left[\frac{1}{1+r_{0,1}^f}\left(\tilde{d}_1 + \tilde{p}_1\right)\right] \qquad (10\text{-}18)$$

implying, in turn, nothing else than $\tilde{x}_t = \tilde{p}_t + \tilde{d}_t$. Equation (10-18) shows that the discounted price plus income process is a martingale. The terms on the left are the price processes, the terms on the right are the conditional expectations under the probability measure Q of the expected cash flows discounted with the risk-free rate (cash flow).

It is important to realise that, with equivalent martingale measures, we deviate from the pure Central Pricing Equation that acknowledges investor-specific discount factors and preferences. From a "true price" perspective, using $\tilde{m}_{0,t} = 1/\left(1 + r_{0,t}^{f}\right)$ would be justified only if there is no risk inherent to x_t (i.e., $\tilde{x}_t = x_t \quad \forall t$), which is only true for very few instruments.

It is the absence of arbitrage principle that forces market prices to be consistent with Equation (10-18). Always keep in mind, however, that this price may not be the best one reflecting your own economic situation and preferences, but rather the one that you might expect to see in the markets as an equilibrium outcome.

The equivalent martingale measure $E^{Q}[]$ is a mathematical construct. It illustrates that the concept of arbitrage depends only on the structure of the price and payoff processes and not on the actual probabilities. As we will see later, equivalent martingale measures simplify the computation of the pricing of derivatives in important ways.

The term "equivalent probability measure" comes from the statistical theory: Given a probability measure P the probability measure Q is said to be equivalent to P if both assign probability zero to the same events. An equivalent probability measure Q is an equivalent martingale measure if all price processes discounted will become martingales. More precisely, Q is an equivalent martingale measure if and only if the market value of any trading strategy is a martingale.

10.5.7 Special Case III: Risk-neutral Probabilities

The concept of equivalent martingale measure requires that the stochastic discount factor $\tilde{m}_{0,t}$ is equal to the risk-free rate $1/1 + r_{0,t}^{f}$. In this section we will have a closer look at this and the conditions under which this makes sense.

With the risk-free rate, the discount factor is state independent. It is common to assume that $\beta = 1/1 + r_{0,t}^{f}$, that the subjective discount factor of the risk-neutral investor is equal to the risk-free rate.

Much of the difficulty in interpreting the Central Pricing Equation lies in the (unknown degree of) risk aversion of individual investors over time and between states of nature. How would the central pricing equation look if investors were not risk-averse?

Recall the definition of the state dependent discount factor from Section 10.5.4:

$$m_{0,t}(\omega) = \frac{q_0(\omega)}{\pi_0(\omega)} = \beta \frac{u'\left(c_t(\omega)\right)}{u'(c_0)}$$

From the discussion of utility functions in Section 10.4.2 we know that if investors are risk-neutral, the first derivative of their utility function is constant for all consumption levels; that is, $u'(c_0) = u'(c_t(\omega))\ \forall\omega$. As a consequence, $u'(c_0)/u'(c_t(\omega)) = 1\ \forall\omega$ and the discount factor becomes state-independent, equal to the subjective discount factor of a risk-neutral investor, independent of consumption levels:

$$m_{0,t}(\omega) = \frac{q_0(\omega)}{\pi(\omega)} = \beta$$

Another useful relationship may be found if we take the right two expressions and rearrange them to have $q_0(\omega) = \pi(\omega)\beta$. As $\sum_{\omega\in\Omega}\beta\pi_0(\omega) = \beta\sum_{\omega\in\Omega}\pi_0(\omega) = \sum_{\omega\in\Omega}q_0(\omega)$ and, by its definition as a probability function $\sum_{\omega\in\Omega}\pi_0(\omega) = 1$ we have

$$E_0\left[m_{0,t}(\tilde{\omega}_t)\right] = \beta = \sum_{\omega\in\Omega} q_0(\omega)$$

Consequently, with risk-neutral investors, the expected value of the discount factor is unique across all states and itself equal to the sum of state prices.

For the case of risk-neutral investors, we denote the discount factor as $m^*_{0,t}(\omega)$ and the risk-neutral probability as $\pi^*_0(\omega)$. Using this and expanding it we find the definition of the risk-neutral probability in terms of discount factors

$$\pi^*_0(\omega) = \frac{q_0(\omega)}{\beta} = \frac{q_0(\omega)}{\sum_{\omega\in\Omega} q_0(\omega)}$$

$\pi^*_0(\omega)$ are positive, less than or equal to one, and sum to one, so they are a legitimate set of probabilities. Using $q_0(\omega_t) = \beta\pi^*_0(\omega)\ \forall\omega$ and $\beta = 1/\left(1 + r^f_{0,t}\right)$, assuming a risk-free asset, we can rewrite the asset pricing formula for risk-neutral investors as

$$p_0 = \sum_{\omega\in\Omega} q_t(\omega)x_t(\omega) = \frac{1}{1+r^f_{0,t}}\sum_{\omega\in\Omega}\pi^*_0(\omega)\,x_t(\omega) = \frac{1}{1+r^f_{0,t}}E^*_0[\tilde{x}_t]$$

which is the formulation we need to justify the equivalent martingale measure in the previous section; that is, $E_t^*[\,] = E_0^Q[\,]$. E_t^* denotes the conditional expectation with respect to the modified probability distribution.

The assumption of risk neutrality of the investor allows eliminating the utility function from the stochastic discount factor. As a consequence, the probabilities $\pi_0^*(\omega)$ differ from the subjective probabilities $\pi_0(\omega)$ in that they are not dependent on a utility function, which is equivalent to assuming that the investor(s) is risk-neutral, hence the name **risk-neutral probabilities**. They are not, in general, the real probabilities associated with states by individual investors.

The use of risk-neutral probabilities has a very fundamental interpretation: risk aversion is equivalent to paying more attention to unpleasant states, relative to their actual probability of occurrence. People who report high subjective probabilities of unpleasant events may not have irrational expectations, but simply may be reporting the risk-neutral probabilities or the product $q_0(\omega) = \pi_0(\omega)m_{0,t}(\omega)$. This product is, after all, the most important piece of information for many decisions: pay a lot of attention to contingencies that are either highly probable or that are improbable but have disastrous consequences!

Bringing back the discount factor and inserting $q_0(\omega) = \pi_0(\omega)m_{0,t}(\omega)$ and $E_0[m_{0,t}(\tilde{\omega}_t)] = \beta = \sum_{\omega\in\Omega} q_0(\omega)$ into the definition of the risk-neutral probability (…) we obtain the transformation from actual (i.e., risk-averse) to risk-neutral probabilities:

$$\pi_0^*(\omega) = \frac{m_{0,t}(\omega)}{E_0\left[m_{0,t}(\tilde{\omega}_t)\right]}\pi_0(\omega)$$

We can also think of the discount factor m as the **derivative** or **change of measure** from the real probabilities π to the subjective probabilities π^*. The risk-neutral probability representation of asset pricing is quite common, especially in derivative pricing where the results are often independent of risk adjustments.

If $\beta = 1/\left(1+r_{0,t}^f\right)$ then we can write the two-dates Central Pricing Equation as

$$p_0^* = \beta E_0^*[\tilde{x}_t] = \frac{1}{1+r_{0,t}^f}E_0^*[\tilde{x}_t]$$

This page intentionally left blank

APPENDIX A

Building Block Data Model

A.1 The Instrument Model

A.1.1 Instrument Core

The *Instrument Core* class of attributes stores information that is generic to all financial instruments. It also provides the central link to the other attribute classes that describe more specific aspects of a financial instrument.

Table A.1: Attribute definitions of the *Instrument Core* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Instrument ID	PK	Unique identifier for this instrument. Automatically allocated by the system.	Identifier	M	(Automatic)
Instrument Status	—	Status of the Instrument: Unissued, Rejected, Issued, Redeemed, etc.	Look-up	M	Instrument Status
Name	—	Name of the instrument.	String	M	—
Primary Asset Class	—	Broad asset classification of the instrument.	Look-up	M	Asset Classification
Denomination Unit	—	Unit in which the instrument is denominated or measured. This is usually NOMINAL, but can also be PERCENT or other denomination units. As compared to the [Quotation Basis] attribute of this class, which specifies the denomination in which the instrument is traded in the markets, this attribute specifies the quantity in which the instrument itself is measured. This may differ from trading conventions for the instrument.	Look-up	M	Unit

Continued

Table A.1: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Min Denomination Amount	—	Defines the smallest amount in which the instrument can be held; i.e., the minimum amount of the instrument in which the buyer is allowed to invest.	Amount	M	—
Nominal Currency	—	Currency in which the instrument is denominated. Also called denomination amount.	Look-up	M	Currency (ISO 4217)
Quotation Basis	—	Defines the default price quotation convention applicable to the instrument; e.g., Percentage-Clean, Percentage-Dirty, Unit-Based, Yield-Based, etc. The purpose of this field is to indicate the default basis for price quotes for the instrument. As compared to the [Denomination Unit] attribute this attribute specifies denomination in which the instrument is traded in the markets.	Look-up	M	Quotation Basis
Day Count Convention	—	Day count convention applicable to the instrument, for example ACT/ACT, 30/360, 30/360E, etc.	Look-up	M	Day Count Convention
Settlement Date Offset Rule	—	Default number of days from the day the trade is done/ issue is placed (T) until the trade is settled. For a standard T+3 settlement, this field should be set to "T+3B" if reported in business days or "T+3C" if reported in calendar days.	Look-up	M	Date Offset Rule

Table A.1: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Date Roll Rule	—	The Date Roll Rule Field specifies the convention to use when determining an appropriate business day in the case that a scheduled event happens to be a nonbusiness day unless adjusted in accordance to the given rule.	Look-up	O	Date Roll Rule
Scheduled Maturity Date	—	Date on which the instrument is scheduled to legally cease to exist.	Date	O	—
Is Valid	—	Signals whether an instance of this class is still currently valid or active.	Indicator	M	—
Valid From Date	—	Defines the date from which the instrument is valid.	Date	O	—
Valid To Date	—	Defines the date until which the instrument is valid.	Date	O	—

SQL Table Generation Code

```
drop table if exists "INSTRUMENT CORE";
create table if not exists "INSTRUMENT CORE"
(
    INSTRUMENT_ID                    int                 not null,
    INSTRUMENT_STATUS                varchar(10)         not null,
    NAME                             text                not null,
    PRIMARY_ASSET_CLASS              varchar(10)         not null,
    DENOMINATION_UNIT                varchar(10)         not null,
    NOMINAL_CURRENCY                 varchar(10)         not null,
    QUOTATION_BASIS                  varchar(10)         not null,
    DAY_COUNT_CONVENTION             varchar(10)         not null,
    SETTLEMENT_DATE_OFFSET_RULE      varchar(10)         not null,
    DATE_ROLL_RULE                   varchar(10),
    "SCHEDULED_MATURITY DATE"        date,
    IS_VALID                         tinyint(1)          not null,
    VALID_FROM_DATE                  datetime,
    VALID_TO_DATE                    datetime,
    primary key (INSTRUMENT_ID),
);
```

A.1.2 Instrument Analytic

The *Instrument Analytic* class allows us to represent analytic values such as estimated prices (as far as not stored in the *Instrument Price* class), accrued income, and other calculation results like Beta, Yield, and Volatility. It is the place where you store calculation results for later reuse.

Table A.2: Attribute definitions of the *Instrument Analytic* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Instrument Analytic ID	PK	Unique identifier for an instance of this class. Automatically allocated by the system.	Identifier	M	(Automatic)
Instrument ID	FK	System identifier of this instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Analytic Scheme	PK	Type (or group) of analytics that an instance of this class represents.	Look-up	M	Analytic Scheme
Analytic Scheme Element	PK	Specifies the specific analytic within an Analytic Scheme.	Look-up	M	Analytic Scheme Element
Analytic Amount	—	Calculated analytic value.	Amount	M	—
Unit	—	Unit of measurement in which the analytic value is denominated.	Identifier	M	Unit
Period Start Date	FK	Start of the period for which the analytic value has been calculated. If it applies to a single point in time, the Period Start attribute should be set to that time.	Date	M	—
Period End Date	—	End of the period for which the analytic value has been calculated. If the analytic is for a single point in time, the Period End attribute should be set to that time and coincides with [Period Start Date].	Date	O	—

SQL Table Generation Code

```
drop table if exists "INSTRUMENT ANALYTIC";
create table if not exists "INSTRUMENT ANALYTIC"
(
    INSTRUMENT_ANALYTIC_ID         int                not null,
    INSTRUMENT_ID                  int,
    ANALYTIC_SCHEME                varchar(10)        not null,
    ANALYTIC_SCHEME_ELEMENT        varchar(10)        not null,
    ANALYTIC_AMOUNT                float(10)          not null,
    UNIT                           varchar(10)        not null,
    PERIOD_START_DATE              datetime           not null,
    PERIOD_END_DATE                datetime           not null,
    primary key (INSTRUMENT_ANALYTIC_ID),
);
```

A.1.3 Instrument Cash Flow Element

The *Instrument Cash Flow Element* class allows us to represent the scheduled cash flow structure of the instrument; for example, any fixed, optional, or residual income or capital redemption cash flows scheduled for the instrument. The *Instrument Cash Flow Element* class is used in conjunction with one or more instances of the *Instrument Cash Flow Schedule* class.

Table A.3: Attribute definitions of the *Instrument Cash Flow Element* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Cash Flow Element ID	PK	Unique identifier for a Cash Flow Element that allows the modelling of complex trees of payoff structures using elementary building blocks. The tag links the Cash Flow Element to other parts of the cash flow structure.	String	M	—
Instrument ID	FK	System identifier of this instrument to which a particular Cash Flow Element belongs.	Identifier	M	Instrument ID

Continued

Table A.3: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Cash Flow Element Type	PK	Specifies the type of cash flow represented by the Cash Flow Element, such as fixed coupon, floating coupon, fixed or floating redemption, and redemption or coupon optionalities, conversion rights, and others.	Look-up	M	Cash Flow Element Type
Period Start Date	PK	Start of the period over which the Cash Flow Element is active; e.g., ■ The start of a coupon period ■ The start of a redemption period ■ The start of an option exercise period If an element is active at only one point in time (e.g., it has a single fixed coupon, redemption, or option exercise date), both Period Start Date and Period End Date should have the same value.	Date	M	—
Period End Date	—	End of the period over which the Cash Flow Element is active, as started by Period Start Date.	Date	O	—
Base Instrument Identifier Type	—	The Base Instrument attribute specifies the currency or other instrument (e.g., gold) in which the cash flow represented by an instance of this class is denominated. This may be the same as the Nominal Currency in the *Instrument Core* class for simple instruments but can be a different instrument for structured and hybrid instruments and need not be a currency. For example, it could be a commodity like gold in the case of a commodity-linked income, redemption or optionality feature of a commodity-linked structured debt or hybrid instrument. Currencies are set up as instruments with Numbering Scheme equal to Currency ISO 3166.	Look-up	M	Numbering Scheme

Table A.3: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Base Instrument Identifier	—	Identifier value corresponding to Base Instrument Identifier Type.	String	M	—
Base Unit	—	Unit on which the Cash Flow Element is based, usually either UNIT or NOTIONAL. NOTIONAL is the normal unit for debt coupons and redemptions and UNIT is the normal unit for equity or fund elements.	Look-up	M	Unit
Base Value	—	Notional or nominal amount on which the element is based per unit of the instrument. Mainly used to specify an issue discount on the Cash Flow Element. To be divided by the Base Value Denominator.	Number	M	—
Base Value Denominator	—	Actual amount paid for the instrument at issue. Used to divide the Base Value. The Base Value Denominator is less than the Base Value if the instrument is issued at a discount and greater than the Base Value if the instrument is issued at a Premium. For redemption Cash Flow Elements this allows us to record the effective actual or implied issue price.	Number	M	—
Rate Constant	—	Fixed pay-off rate for the Cash Flow Element. For debt coupons this represents ■ The coupon rate of fixed coupon bonds ■ The fixed spread in the case of floaters For debt redemptions this is the rate of the Base Unit to be redeemed not conditional on any other terms. This is normally 1.0 (or 100 %) but may be higher or lower.	Rate	M	—

Continued

Table A.3: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Rate Factor	—	Multiplier for any variable part of the overall payoff defined by the Cash Flow Element. For debt coupons this is the multiplier by which the underlying rate is to be adjusted, often set to 1.0. See Underlying related attribute for the identification of the underlying rate. For debt redemptions this is the multiplier by which the underlying rate is to be adjusted to arrive at the variable part of the redemption (if any). Set to 0 (or 0%) in the case if there is no variable part. For variable redemption structures it may be 1.00 (or 100%) of a cumulative return reference instrument. If this is the case the Index Base constant will need to be set to take into account initial cumulative of the underlying.	Number	O	—
Cap	—	Maximum payoff rate for the Cash Flow Element. Used in conjunction with Rate Constant and Rate Factor.	Amount	O	—
Floor	—	Maximum payoff rate for the Cash Flow Element. Used in conjunction with Rate Constant and Rate Factor.	Amount	O	—
Underlying Instrument Identifier Type	—	Specifies the instrument used as reference for any variable payoff components. For Floating Rate Coupons or Variable Redemption elements this is the instrument on which the variable rate or redemption will be based.	Look-up	O	Numbering Scheme
Underlying Instrument Identifier	—	Identifier value for Underlying Instrument Identifier Type.	String	O	—

Table A.3: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Underlying Type	—	Specifies whether the underlying is a single instrument, a portfolio, or if the underlying is a set of child elements to be chained together in sequence.	Look-up	O	Underlying Type
Underlying Price Type	—	Price type of the reference series to be used for fixing the payoff entitlement for the Cash Flow Element. This may be closing price, interest rate, cumulative value or return, variance, etc., as appropriate.	Look-up	O	Price Type

SQL Table Generation Code

```
drop table if exists "INSTRUMENT CASH FLOW ELEMENT";
create table if not exists "INSTRUMENT CASH FLOW ELEMENT"
(
    CASH_FLOW_ELEMENT_ID                int              not null,
    INSTRUMENT_ID                       int              not null,
    CASH_FLOW_ELEMENT_TYPE              varchar(10)      not null,
    PERIOD_START_DATE                   datetime         not null,
    PERIOD_END_DATE                     datetime,
    BASE_INSTRUMENT_IDENTIFIER_TYPE varchar(10)          not null,
    BASE_INSTRUMENT_IDENTIFIER          varchar(20)      not null,
    BASE_UNIT                           varchar(10)      not null,
    BASE_VALUE                          varchar(10)      not null,
    BASE_VALUE_DENOMINATOR              varchar(10)      not null,
    RATE_CONSTANT                       numeric(10,6)    not null,
    RATE_FACTOR                         numeric(10,6),
    CAP                                 numeric(10,6),
    FLOOR                               numeric(10,6),
    UNDERLYING_INSTRUMENT_IDENTIFIER_TYPE varchar(10),
    UNDERLYING_INSTRUMENT_IDENTIFIER varchar(20),
    UNDERLYING_TYPE                     varchar(10),
    UNDERLYING_PRICE_TYPE               varchar(10),
    primary key (CASH_FLOW_ELEMENT_ID),
);
```

A.1.4 Instrument Cash Flow Schedule

The *Instrument Cash Flow Schedule* class specifies the first and last dates on which a particular cash flow event for a Cash Flow Element is planned to occur, as well as how to determine any future regular recurrences of that event.

Table A.4: Attribute definitions of the *Instrument Cash Flow Schedule* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Cash Flow Schedule ID	PK	Unique identifier for an instance of this class. Automatically allocated by the system.	Identifier	M	(Automatic)
Cash Flow Element ID	FK	Identifies Cash Flow Element to which an instance of this class belongs.	String	M	—
Cash Flow Schedule Type	—	Type of event the series of which is modelled by the instance of the Cash Flow Schedule class; e.g., accrual date, settlement date, fixing date, etc.	Look-up	M	Cash Flow Schedule Type
Repetition Period Type	—	Specifies the type of period defined for the repetition of the event; e.g., Day, Week, Month, Year.	Look-up	O	Repetition Period Type
Repetition Period Length	—	Specifies the length of the interval between successive repetitions of the event in terms of the defined period type; e.g., 2 (days), 3 (months), etc.	Number	O	—
Schedule Start Date	—	Specifies the first occurrence of the Cash Flow Element and the start of the repetition sequence if the Cash Flow Element is recurring.	Date	M	—
Schedule End Date	—	Specifies the last occurrence of the Cash Flow Element if the Cash Flow Element is recurring. If it is not recurring it must be set to the same date as the Schedule Start Date attribute.	Date	M	—

SQL Table Generation Code

```
drop table if exists "INSTRUMENT CASH FLOW SCHEDULE";
create table if not exists "INSTRUMENT CASH FLOW SCHEDULE"
(
    CASH_FLOW_SCHEDULE_ID        int              not null,
    CASH_FLOW_ELEMENT_ID         int              not null,
    INSTRUMENT_ID                int              not null,
    CASH_FLOW_SCHEDULE_TYPE      varchar(10)      not null,
    REPETITION_PERIOD_TYPE       varchar(10),
    REPETITION_PERIOD_LENGTH     varchar(10),
    SCHEDULE_START_DATE          datetime         not null,
    SCHEDULE_END_DATE            datetime         not null,
    primary key (CASH_FLOW_SCHEDULE_ID),
);
```

A.1.5 Instrument Cash Flow Fixing

The *Instrument Cash Flow Fixing* class of attributes allows the representation of actual fixings that have taken place, defining the actual values of any variable coupon, redemption, or cash flow related parameters such as pool factors for any instrument.

Table A.5: Attribute definitions of the *Instrument Cash Flow Fixing* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Cash Flow Fixing ID	PK	Unique identifier for an instance of this class. Automatically allocated by the system.	Identifier	M	(Automatic)
Instrument ID	FK	Unique identifier for this instrument data record. This attribute specifies the instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Cash Flow Element ID		Specifies the element to which an instance of this class belongs.	String	O	—
Cash Flow Fixing Type		Specifies the type of instrument element to which the fixing value applies.	Look-up	M	Cash Flow Fixing Type

Continued

Table A.5: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Fixing Date		Specifies the date on which the actual fixing has taken place.	Date	O	—
Effective Date		Specifies the date from which onward the fixed value applies.	Date	M	—
Fixing Value		Specifies the actual value determined for the fixing.	Amount	M	—
Fixing Unit		Specifies the unit in which the fixing is expressed.	Look-up	M	Unit

SQL Table Generation Code

```
drop table if exists "INSTRUMENT CASH FLOW FIXING";
create table if not exists "INSTRUMENT CASH FLOW FIXING"
(
    CASH_FLOW_FIXING_ID        int               not null,
    CASH_FLOW_ELEMENT_ID       int               not null,
    INSTRUMENT_ID              int               not null,
    CASH_FLOW_FIXING_TYPE      varchar(10)       not null,
    FIXING_DATE                datetime,
    EFFECTIVE_DATE             datetime          not null,
    FIXING_VALUE               numeric(15,6)     not null,
    FIXING_UNIT                varchar(10)       not null,
    primary key (CASH_FLOW_FIXING_ID),
);
```

A.1.6 Instrument Identifier

The *Instrument Identifier* class or attributes captures information on the instrument identification. Data on multiple identifiers for the same instrument can be stored in a repetitive mode.

Table A.6: Attribute definitions of the *Instrument Identifier* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Instrument Identifier ID	PK	Unique identifier for an instance of this class. Automatically allocated by the system.	Identifier	M	(Automatic)
Instrument ID	FK	System identifier of this instrument to which an instance of this class belongs.	Identifier	M	—
Numbering Scheme	—	Specifies the identifier coding scheme or numbering scheme according to which a value for this financial instrument is stored. The numbering scheme can be, for example, ISIN.	Look-up	M	Numbering Scheme
Identifier Value	—	Stores the actual value of the identifier, for example the ISIN code for this financial instrument.	String	M	—

SQL Table Generation Code

```
drop table if exists “INSTRUMENT IDENTIFIER”;
create table if not exists “INSTRUMENT IDENTIFIER”
(
    INSTRUMENT_IDENTIFIER_ID      int            not null,
    INSTRUMENT_ID                 int            not null,
    NUMBERING_SCHEME              varchar(10)    not null,
    IDENTIFIER_VALUE              varchar(20)    not null,
    primary key (INSTRUMENT_IDENTIFIER_ID),
);
```

A.1.7 Instrument Income Payment

The *Instrument Income Payment* class records the history of actual income payments for an instrument. This includes declared and actually settled dividend payments as well as actual coupon payments. For the purpose of this book, we abstract from noncash income as well as any tax considerations.

Table A.7: Attribute definitions of the *Instrument Income Payment* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Income Payment ID	PK	Holds the unique system internal identifier for the record.	Identifier	M	—
Instrument ID	FK	System identifier of this instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Income Event Type	—	Specifies the type of income event that leads to the income arising. Determines whether the corporate action event relates to an actual coupon payment (INTR) the announcement of the coupon fixing for a floating rate instrument (CPNR).	Identifier	M	Income Event Type
Period Start Date	—	Specifies the start of the period to which the income payment relates. For debt instruments this is the start date of the coupon period to which the coupon payment or fixing refers. For shares this is the start date of the period to which the dividend payment relates.	Date	O	—
Period End Date	—	Specifies the end of the period started by Period Start Date.	Date	O	—
Income Accrued in Cash	—	Specifies the actual value of the income in terms of the Nominal Currency.	Amount	M	—

SQL Table Generation Code

```
drop table if exists "INSTRUMENT INCOME PAYMENT";
create table if not exists "INSTRUMENT INCOME PAYMENT"
(
    INCOME_PAYMENT_ID           int              not null,
    INSTRUMENT_ID               int              not null,
    INCOME_EVENT_TYPE           varchar(10)      not null,
    PERIOD_START_DATE           datetime,
    PERIOD_END_DATE             datetime,
    INCOME_ACCRUED_IN_CASH      numeric(10,6)    not null,
    primary key (INCOME_PAYMENT_ID),
);
```

A.1.8 Instrument Index Version

The *Instrument Index Version* and *Instrument Index Constituent* attribute classes allow us to represent Index definitions and versions of these definitions such as Equity or Bond Indices.

The *Instrument Index Version* allows us to record a unique set of parameters and constituents applicable for a defined period. If the constituents or parameters of an index are changed, a new version of the index is created.

Table A.8: Attributes of the *Instrument Index Version* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Index Version ID	PK	Holds the unique system internal identifier for the record.	Identifier	P	—
Instrument ID	FK	System identifier of this instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Index Valuation Formula	—	Specifies the formula used to calculate the index value.	Look-up	M	Index Valuation Formula
Index Valuation Variable	—	Specifies the variable used for representing the value of each index component.	Look-up	M	Index Valuation Variable
Index Weighting Variable	—	Specifies the variable used for representing the weight of each index component.	Look-up	M	Index Weighting Variable

Continued

Table A.8: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Initial Amount	—	Specifies the initial value of the index version. Allows normalisation of the index at inception to a certain value.	Amount	M	—
Is Market Price Required	—	Specifies whether market prices are required for this index or whether evaluated prices are acceptable.	Indicator	M	—
Minimum Constituents	—	Specifies the minimum number of constituents required for a valid index calculation.	Amount	M	—
Minimum Days Near Maturity Amount	—	Specifies the minimum remaining life for an instrument that can be included in the index.	Amount	M	—
Maximum Missing Price Threshold	—	Specifies the maximum age of prices in calendar days to be used in the index calculation.	Amount	M	—
Chaining Factor	—	Specifies the multiplier that chains together the current version with its predecessor. It captures the multiplier to link the last calculated value of the previous index version to the first calculated value of the current index version.	Amount	M	—
Is Official	—	Signals whether an instance of this class is from an official source or deemed to be official.	Indicator	M	—
Is Valid	—	Signals whether the information provided in this record is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	M	—

Table A.8: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Valid From Date	—	Defines the date from which the information provided in this record is valid for the financial instrument.	Date	M	—
Valid To Date	—	Defines the date until which the information provided in this record is valid for the financial instrument.	Date	O	—

A.1.9 Instrument Index Constituent

The *Instrument Index Constituent* class is used to record the financial instruments that are used for an index. Details on how to set up custom indices are given in Chapter 9.

Table A.9: Attributes of the *Instrument Index Constituent* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Index Constituent Identifier	PK	Holds the unique system internal identifier for the record.	Identifier	M	N/A
Instrument ID	FK	System identifier of this instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Index Version ID	FK	Specifies the index version to which this constituent belongs.	Identifier	M	Instrument Index Version ID
Weight Factor	—	Specifies the weighting to be given to the instrument within the index.	Amount	M	N/A
Chaining Factor	—	Specifies the multiplier to be used to adjust the value of this constituent before it is used in the index calculation. This normally is used to adjust for example for dividends and stock split.	Amount	M	N/A

Continued

Table A.9: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Is Valid	—	Signals whether the information provided in this record is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	M	N/A
Valid From Date	—	Defines the date from which the information provided in this record is valid for the financial instrument.	Date	O	N/A
Valid To Date	—	Defines the date until which the information provided in this record is valid for the financial instrument.	Date	O	N/A

A.1.10 Instrument Issuance

The *Instrument Issuance* class of attributes represents the history of Issuance events for an instrument. We do not enter into the intricacies of various offer types, and issuance statuses or issuance events. We only record issuance data for instruments that have actually been issued and record the factual issuance data.

Table A.10: Attributes of the *Instrument Issuance* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Instrument Issuance ID	PK	Unique system internal identifier for any instance of this class.	Identifier	P	N/A
Instrument ID	FK	Unique identifier for this instrument data record. This attribute specifies the instrument to which an instance of this class belongs.	Identifier	M	Instrument ID

Table A.10: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Issuance Transaction Type		Specifies the type of the transaction; e.g., IPO, SPO, Redenomination, Stock Split, Buy-Back, etc.	Identifier	M	Issuance Transaction Type
Number Of Tranches		Used in case of multitranche offering to indicate the number of separately and explicitly defined tranches. This applies, for example, to retail offering, institutional offering, domestic offering, etc.	Amount	O	N/A
Name Of Tranche		Used in case of multitranche offering, where it captures the actual name of this tranche or the Tranche ID. This attribute identifies the tranche (in case of multiple tranches) or the program to which the issuance is related. In case that multiple issuance transactions take place on the same date, this number tracks them separately.	String	O	N/A
Issuance Paid Amount		Specifies the actual change in the overall amount issued due to this issuance transaction. In the case of issuance transactions increasing the overall amount outstanding, this attribute is positively defined. In case of a buy-back decreasing the overall amount issued, this amount is negatively defined.	Amount	O	N/A

Continued

Table A.10: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Issuance Amount Currency		Specifies the currency on which the Issuance Paid Amount and the Issuance Offer Amount are expressed.	Identifier	M	Currency (ISO 4217)
Issuance Paid Number		Specifies the number of units issued in this issuance transaction. To be filled in only if the record corresponds to a transaction where the company issues/creates primary shares (as opposed to a sale of secondary/existing shares by an existing shareholder). For some transactions (e.g., buy-backs or reverse stock splits like a 2 to 1 where 2 old shares become 1 new share), this field could contain negative values.	Amount	O	N/A
Issue Price		Specifies the total value of all consideration physical and cash ultimately paid for each unit of the instrument, expressed in terms of the Consideration Currency. This is the final price for which the units of the instrument ultimately are sold.	Amount	O	N/A
Consideration Currency		Also called Issue Price Currency; specifies the currency in which any consideration for the issuance transaction is paid. Currency in which the transaction is settled.	Identifier	O	Currency (ISO 4217)

Table A.10: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Consideration Cash		The part of the issue price that is paid in cash (equal to total issue price in the case that no physical payment is made). The Consideration Cash is any net cash amount paid or received by the investor for the issuance transaction.	Amount	O	N/A
Consideration Physical		The part of the issue price that is paid in noncash (e.g., delivery of securities).	Amount	O	N/A
Consideration Instrument ID		Specifies the instrument in which the consideration physical is denominated (instrument ID for any instrument used as physical payment).	Identifier	O	Instrument ID
Accrued Coupon At Issue		Accrued coupon for securities (in %) as a decimal number corresponding to the percentage of nominal. This attribute applies where Accrual Start Date is prior to the Issue Date.	Number	O	N/A

A.1.11 Instrument Issuance Date

The *Instrument Issuance Date* class of attributes represents the set of dates applicable to the history of issuance transactions recorded for an instrument.

Table A.11: Attributes of the *Instrument Issuance* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Issuance Date ID	PK	Unique system internal identifier for any instance of this class.	Identifier	P	N/A

Continued

Table A.11: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Instrument ID	FK	Unique identifier for this instrument data record. This attribute specifies the instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Instrument Issuance ID	—	Unique system internal identifier for any instance of the Instrument Issuance class.	Identifier	M	Issuance ID
Issuance Date Type	—	Specifies the stage of the issuance process to which the date specified in Issuance Date refers.	Date	O	Issuance Date Type
Issuance Date	—	Specifies the date corresponding to the stage of the issuance process defined in Issuance Date type.	Date	O	N/A

A.1.12 Instrument Rating

The *Instrument Rating* class of attributes may be used to represent the various proprietary and public ratings that are assigned to instruments and reviewed from time to time. This includes credit ratings for bonds as well as more general risk ratings.

Table A.12: Attribute definitions of the *Instrument Rating* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Instrument Rating ID	PK	Unique system internal identifier for any instance of this class.	Identifier	M	—
Instrument ID	FK	System identifier of this instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Rating Scheme	—	Specifies the rating agency and the type of rating to which the Rating Scheme Value corresponds; e.g., Moody's Short Term Rating, etc.	Look-up	M	Rating Scheme

Table A.12: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Rating Scheme Value	—	Specific rating given to the instrument.	String	M	
Is Valid	—	Indicates whether an instance of this class is still currently valid or active.	Indicator	M	—
Valid From Date	—	Date of assignment of this rating. Defines the date from which an instance of this class is valid in the real world (date as of which the information provided is valid for the financial instrument).	Date	O	—
Valid To Date	—	Date on which the assignment of this rating ends (because the rating has been changed). Defines the date until which an instance of this class is valid in the real world (date until when the information provided is valid for the financial instrument).	Date	O	—

SQL Table Generation Code

```
drop table if exists "INSTRUMENT RATING";
create table if not exists "INSTRUMENT RATING"
(
    INSTRUMENT_RATING_ID        int             not null,
    INSTRUMENT_ID               int,
    RATING_SCHEME               varchar(10)     not null,
    RATING_SCHEME_VALUE         varchar(20)     not null,
    IS_VALID                    tinyint(1)      not null,
    VALID_FROM_DATE             datetime,
    VALID_TO_DATE               datetime,
    primary key (INSTRUMENT_RATING_ID),
    unique (),
);
```

A.1.13 Instrument Relationship Condition

The *Instrument Relationship Condition* class represents constraints in terms of relationships between the financial instrument concerned and other financial instruments. Examples are the linkage of benchmarks to portfolios, programmes of short term debt issuances and derivative contract specifications from which actual contracts are derived. Actual Instruments derived from these should then follow the patterns defined in this way.

Table A.13: Attributes of the *Instrument Relationship Condition* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Instrument Relationship Condition ID	PK	The Instrument Relationship Condition Identifier holds the unique system internal identifier for the record.	Identifier	**P**	N/A
Instrument ID	FK	The Instrument Identifier species the financial instrument for which the relationship condition is defined.	Identifier	M	Instrument ID
Condition Scheme	—	The Relationship Condition Scheme attribute specifies the Condion Scheme from which the Condition Scheme Function below is referenced. From the Condition Scheme code list only Condition Schemes with Scheme Type = “RELSHP” are admitted.	Identifier	M	Condition Scheme

Table A.13: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Condition Scheme Function	—	The Condition Scheme Function attribute specifies the function or purpose of the constraint. It can specify the format or domain in which the attributes Condition Minimum and Condition Maximum are formulated.	Identifier	M	Condition Scheme Function
Condition Relationship Type	—	The Condition Relationship Type attribute specifies the relationship type applicable to the condition, e.g., the type of relationship required.	String	M	Condition Relationship Type
Condition Instrument ID	—	The Condition Instrument Identifier specifies for the instrument prototype (template) a specific instrument either as the default instrument or as the required instrument for the specified relationship. For any instruments derived from the instrument prototype for which an instance of this class is defined are required to have a relationship of the given type and characteristics with.	Identifier	O	Instrument ID

Continued

Table A.13: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Condition Instrument Role Type	—	The Condition Instrument Role Type attribute specifies the role an instrument must have in order to be eligible for the required relationship. For example, this attribute may specify that the Instrument entering a relationship as underlying must be a specific Interbank Rate, e.g., 3 month LIBOR.	Identifier	O	Condition Instrument Role Type
Condition Minimum	—	The Condition Minimum attribute specifies the minimum number of active relationships of the given type required.	Amount	O	N/A
Condition Maximum	—	The Condition Maximum attribute specifies the maximum number of active relationships allowed.	Amount	O	N/A
Constraint Type	—	The Constraint Type attribute specifies the precise type of constraint to be set. The kind of constraints can be for instance "REQUIRED—NO DEFAULT", "REQUIRED—WITH DEFAULT", etc.	Identifier	M	Constraint Type
Is Valid	—	The Is Valid attribute indicates whether an instance of this class is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	O	N/A

Table A.13: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Valid From Date	—	The Valid From attribute defines the date from which an instance of this class is valid. Date as of which the information provided is valid for the financial instrument. Start date of the current version of the financial instrument.	Date	O	N/A
Valid To Date	—	The Valid To attribute defines the date until which an instance of this class is valid. Date until when the information provided is valid for the financial instrument. End date of the current version of the financial instrument.	Date	O	N/A

A.2 The Portfolio Model

A.2.1 Portfolio Version

The *Portfolio Version* and *Portfolio Constituent* attribute classes allow us to store information on portfolios of instruments.

The *Portfolio Version* records a unique set of parameters and constituents applicable for a defined period. Portfolio versions may be used if there are changing parameters or if constituent sets are to be frozen.

Table A.14: Attributes of the *Portfolio Version* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Portfolio Version ID	PK	Unique system internal identifier for any instance of the Instrument Curve Version class.	Identifier	**P**	—
Instrument ID	FK	System identifier of this instrument to which an instance of this class belongs.	Identifier	M	Instrument ID

Continued

Table A.14: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Portfolio Version Scheme	—	Specifies the purpose for which the portfolio version is being used.	Identifier	O	Portfolio Version Scheme
Is Valid	—	Signals whether the information provided in this record is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	M	—
Valid From Date	—	Defines the date from which the information provided in this record is valid for the financial instrument.	Date	O	—
Valid To Date	—	Defines the date until which the information provided in this record is valid for the financial instrument.	Date	O	—

SQL Table Generation Code

```
drop table if exists "PORTFOLIO VERSION";
create table if not exists "PORTFOLIO VERSION"
(
    PORTFOLIO_VERSION_ID            int               not null,
    INSTRUMENT_ID                   int,
    PORTFOLIO_VERSION_SCHEME        varchar(10)       not null,
    IS_VALID                        tinyint(1)        not null,
    VALID_FROM_DATE                 datetime,
    VALID_TO_DATE                   datetime,
    primary key (PORTFOLIO_VERSION_ID),
    unique (),
);
```

A.2.2 Portfolio Constituent

The *Portfolio Constituent* class of attributes is used to record the financial instruments that compose a portfolio. See Chapter 9 for details on how to set up portfolios.

Table A.15: Attributes of the *Portfolio Constituent* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Constituent Instrument ID	PK	Unique system internal identifier for constituent records.	Identifier	M	—
Portfolio Version ID	FK	Specifies the portfolio version for which this constituent is used.	Identifier	M	Portfolio Version ID
Instrument ID	FK	System identifier of this instrument to which an instance of this class belongs.	Identifier	M	Instrument ID
Weight Factor	—	Defines the weight of the financial instrument within the portfolio.	Amount	M	—
Effective Date	—	Defines the date from which onward the constituent is active within the portfolio version.	Date	M	—
Constituent Function	—	Specifies the purpose for which this constituent has been recorded.	Identifier	M	Constituent Function
Base Cost Amount	—	Defines the value of the instrument constituent as a chosen (normalising) base price.	Amount	O	—
Base Cost Currency	—	Currency that is used for the Base Cost Amount.	Identifier	O	Currency (ISO 4217)
Is Valid	—	Signals whether the information provided in this record is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	M	—
Valid From Date	—	Defines the date from which the information provided in this record is valid for the financial instrument.	Date	O	—
Valid To Date	—	Defines the date until which the information provided in this record is valid for the financial instrument.	Date	O	—

SQL Table Generation Code

```
drop table if exists "PORTFOLIO CONSTITUENT";
create table if not exists "PORTFOLIO CONSTITUENT"
(
    CONSTITUENT_INSTRUMENT_ID       int                 not null,
    PORTFOLIO_VERSION_ID            int                 not null,
    INSTRUMENT_ID                   int,
    WEIGHT_FACTOR                   numeric(10,2)       not null,
    EFFECTIVE_DATE                  datetime            not null,
    CONSTITUENT_FUNCTION            varchar(10)         not null,
    BASE_COST_AMOUNT                numeric(10,6),
    BASE_COST_CURRENCY              varchar(10),
    IS_VALID                        tinyint(1)          not null,
    VALID_FROM_DATE                 datetime,
    VALID_TO_DATE                   datetime,
    primary key (CONSTITUENT_INSTRUMENT_ID),
    unique (),
    unique (),
    ,
);
```

A.2.3 Portfolio Position

The *Portfolio Position* class of attributes is used to record valuations, risk and return measures for financial instruments that compose a portfolio. See Chapter 9 for details on how to model portfolios.

Table A.16

Attribute Name	PK/FK	Description	Type	M/O	Domain
Portfolio Instrument	PK	ID of the portfolio.	NUMBER	M	—
Portfolio Version	FK	ID of the portfolio snapshot or version.	VARCHAR2	M	—
Constituent Instrument	FK	ID of the instrument held in the portfolio.	NUMBER	M	—
Quantity	—	Long (+) or short (–) position held.	NUMBER		—
Valuation	—	This is either the cost, valuation, or period return for the position.	NUMBER		—
Effective Date	—	Date of last revision to the position.	DATE		—

Table A.16: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Position Transaction Reference	—	Cross reference to the transaction that established the position.	NUMBER		—
Position Function	—	Type of position.	VARCHAR2		—
Position Stage	—	Type of position.	VARCHAR2		—
Currency	—	Currency of the position.	VARCHAR2		—
Is Valid	—	Indicates whether the position is still in use for current or historical reporting or has been entirely superseded.	NUMBER		—
Valid From Date	—	Date from which the position existed or the beginning of a period from which to measure the return on the position.	DATE		—
Valid To Date	—	Last date on which the position existed or the end of a period from which to measure the return on the position.	DATE		—

A.2.4 Benchmark Component

The *Benchmark Component* class of attributes is used to store information on portfolio benchmarks that is to be mapped to portfolios via the Instrument Relationship Condition Class.

Table A.17

Attribute Name	PK/FK	Description	Type	M/O	Domain
Benchmark Instrument	PK	Id of the Benchmark.	NUMBER	M	—
Benchmark Version	FK	Id of the Benchmark Snapshot or Version.	VARCHAR2	M	—
Component Instrument	—	Id of the Instrument held in the benchmark.	NUMBER		—
Weight Factor	—	Weight of the component in the benchmark.	NUMBER		—

Continued

Table A.17: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Return Factor	—	Return of the component over a given period.	NUMBER		—
Chaining Factor	—		NUMBER		—
Effective Date	—	Date of last revision to the position.	DATE		—
Position Function	—	Type of position.	VARCHAR2		—
Position Stage	—	Type of position.	VARCHAR2		—
Currency	—	Currency of the Benchmark Component.	VARCHAR2		—
Is Valid	—	Indicates whether the position is still in use for current or historical reporting or has been entirely superseded.	NUMBER		—
Valid From Date	—	Date from which the position existed or the beginning of a period for which to measure the return on the position.	DATE		—
Valid To Date	—	Last date on which the position existed or the end of a period for which to measure the return on the position.	DATE		—

A.3 The Party Model

A.3.1 Party Core

The *Party Core* class stores information about organisations. It refers to organisations participating in operations with financial instruments acting as one or more of the following roles: issuer, issuer parent organisation, guarantor, depository, exchange, data source, and so on.

Table A.18: Attribute definitions of the *Party Core* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Party ID	PK	Unique system identifier for any instance of this class.	Identifier	M	—
Name	—	Stores the actual name.	String	M	—

Table A.18: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Party Type	—	Specifies the form of the party; e.g., company or other commercial legal entity, sovereign government, unincorporated division, etc.	Look-up	M	Party Type
Current Status	—	Specifies the present status of the party; e.g., active, dissolved, etc.	Look-up	M	Party Status
Domicile Country	—	Specifies the country in which the party is domiciled. Domicile is the country of registration/incorporation.	Look-up	M	Country (ISO 3166)
Registration Place	—	Specifies the name and (possibly) location of the registration authority or jurisdiction of the party. In the case of most European countries this is the name of the authority in combination with the name of the city in which the company has been registered (e.g., for Germany: *Amtsgericht Darmstadt*). For the UK, this would be representing the jurisdiction applying, such as “England & Wales” for an English or Welsh company, “Scotland” for a Scottish company.	String	O	—
Base Currency	—	Specifies the primary currency in which the party’s accounting records are denominated.	Look-up	O	Currency (ISO 4217)
Is Valid	—	Indicates whether an instance of this class is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	M	—
Valid From Date	—	Defines the date from which an instance of this class is valid.	Date	O	—
Valid To Date	—	Defines the date until which an instance of this class is valid.	Date	O	—

SQL Table Generation Code

```
drop table if exists "PARTY CORE";
create table if not exists "PARTY CORE"
(
    PARTY_ID            int             not null,
    NAME                text            not null,
    PARTY_TYPE          varchar(10)     not null,
    CURRENT_STATUS      varchar(10)     not null,
    BASE_CURRENCY       varchar(10),
    IS_VALID            tinyint(1)      not null,
    VALID_FROM_DATE     datetime,
    VALID_TO_DATE       datetime,
    primary key (PARTY_ID)
);
```

A.3.2 Party Account

The package *Party Account* represents financial profiles and reports for parties. This includes income, balance sheet, and flow of funds statements, as well as lower level analytical accounts. The Party Account class stores financial profiles and reports for parties. This includes income, balance sheet, and flow of funds statements as well as lower level analytical accounts.

Table A.19: Attribute definitions of the *Party Account* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Party Account ID	PK	Unique system internal identifier for any instance of this class.	Identifier	P	—
Party ID	FK	Specifies the party to which an instance of this class belongs.	Identifier	M	Party ID
Party Account Scheme	—	Specifies the scheme from which the Party Account Scheme Value is selected.	Look-up	O	Party Account Scheme

Table A.19: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Party Account Scheme Element	—	Specifies the function or purpose of the amount within a financial report that an instance of this class represents. For example, this could indicate that the amount is Shareholders Equity in a balance sheet report or Gross Revenue from Sales in an income statement.	Look-up	M	Party Account Scheme Element
Party Account Amount Function	—	Indicates the function of the party account amount(s). This can be, for example, End of Year Balance, Half Year Balance, Quarterly Balance, Month End Balance, Account Movement, etc.	Look-up	M	Party Account Amount Function
Party Account Status	—	Specifies the present status of the Party Account. For instance, this could be OPEN, RECONCILED, CLOSED, etc.	Look-up	M	Party Account Status
Original Currency	—	Specifies the currency in which the Original Currency Amount is denominated.	Look-up	M	Currency (ISO 4217)
Original Currency Amount	—	Stores the actual value of the stock or flow amount in the given financial report in the original currency. The original currency of record is the transaction currency behind.	Amount	M	—
Base Currency Amount	—	Records the stock or flow amount expressed in terms of the base currency on which the accounting records of the party are denominated.	Amount	M	—

Continued

Table A.19: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Period Start Date	—	Defines the start of the period to which the party account amount(s) applies, for period based statements. If the party account amount(s) refer to a single point in time rather than a period, the Period Start attribute should be set to the same value as Period End.	Date	M	—
Period End Date	—	Defines the end of the period to which the party account amount(s) apply, for period based statements. If the party account amount(s) refers to a single point in time rather than a period, the Period End attribute should be set to that date.	Date	M	—
Narrative	—	Allows storing an optional short comment for the party account amount.	String	O	—
Debit Credit	—	Specifies whether the party account amount is a debit or credit entry.	Look-up	M	Debit Credit
Is Valid	—	Indicates whether an instance of this class is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	M	—

SQL Table Generation Code

```
drop table if exists "PARTY ACCOUNT";
create table if not exists "PARTY ACCOUNT"
(
    PARTY_ACCOUNT_ID                 int                not null,
    PARTY_ID                         int                not null,
    "PARTY ACCOUNT SCHEME"           varchar(10)        not null,
    PARTY_ACCOUNT_SCHEME_ELEMENT     varchar(10)        not null,
    PARTY_ACCOUNT_AMOUNT_FUNCTION    varchar(10)        not null,
    PARTY_ACCOUNT_STATUS             varchar(10)        not null,
    ORIGINAL_CURRENCY                varchar(10)        not null,
    ORIGINAL_CURRENCY_AMOUNT         numeric(10,2)      not null,
    BASE_CURRENCY_AMOUNT             numeric(10,2)      not null,
    PERIOD_START_DATE                datetime           not null,
    PERIOD_END_DATE                  datetime           not null,
    NARRATIVE                        varchar(255),
    DEBIT_CREDIT                     varchar(10)        not null,
    "IS VALID"                       tinyint(1)         not null,
    primary key (PARTY_ACCOUNT_ID),
);
```

A.3.3 Party Analytic

The *Party Analytic* package contains the class of attributes needed to represent imported or calculated analytic values for a party, such as earnings per share and others. The Party Analytic class stores imported, derived, or calculated analytic values for a party. It stores information that can represent simple scalar analytics, such as volatility, as well as information for curve-based analytics. Curve-based analytics include, for example, default curves in the case of funds or accounting, performance, and financial ratios for other issuers.

Table A.20: Attribute definitions of the *Party Analytic* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Party Analytic ID	PK	Unique system internal identifier for any instance of this class.	Identifier	P	—
Party ID	FK	Specifies the party to which an instance of this class belongs.	Identifier	M	Party ID

Continued

Table A.20: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Analytic Scheme	—	Specifies the category of analytics to which an instance of this class belongs such as Accounting analytics or Volatility.	Look-up	M	Analytic Scheme
Analytic Scheme Element	—	Specifies the precise type of analytic that an instance of this class represents. This could be a ratio such as Interest Cover, Dividend Cover, Quick Ratio, or in the case of a fund, the volatility of the underlying fund portfolio.	Look-up	M	Analytic Scheme Element
Analytic Amount	—	Records the actual calculated analytic value.	Amount	M	—
Calculation Date	—	Specifies the date and time at which the Analytic Amount was calculated.	Date	M	—
Unit	—	Defines the unit of measurement in which the Analytic Amount is denominated.	Look-up	M	Unit
Currency	—	Specifies the currency in which the Analytic Amount is denominated, if applicable.	Look-up	O	Currency (ISO 4217)
Period Start Date	—	Defines the start of the period from which the analytic value has been calculated. If the analytic is for a single point in time rather than a period the Period Start attribute should be set to that time. In the case of funds examples are: ■ Effective date of total assets under management ■ Start date of the period to which the turnover ratio refers	Date	M	—

Table A.20: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Period End Date	—	Defines the end of the period from which the analytic value has been calculated. If the analytic is for a single point in time rather than a period the Period End attribute should be set to that time.	Date	O	—
Is Valid	—	Indicates whether an instance of this class is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	M	—

SQL Table Generation Code

```
drop table if exists "PARTY ANALYTIC";
create table if not exists "PARTY ANALYTIC"
(
    PARTY_ANALYTIC_ID            varchar(10)      not null,
    PARTY_ID                     int              not null,
    ANALYTIC_SCHEME              varchar(10)      not null,
    ANALYTIC_SCHEME_ELEMENT      float(10)        not null,
    CALCULATION_DATE             datetime         not null,
    UNIT                         varchar(10)      not null,
    CURRENCY                     varchar(10),
    PERIOD_START_DATE            datetime         not null,
    PERIOD_END_DATE              datetime,
    "IS VALID"                   varchar(10)      not null,
    primary key (PARTY_ANALYTIC_ID),
);
```

A.3.4 Party Industry Classification

The *Party Industry Classification* allows us to store information about the party's economic activity.

Table A.21: Attribute definitions of the *Party Industry Classification* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Party Industry ID	PK	Unique system internal identifier for this class.	Identifier	M	—
Party ID	FK	System identifier for the party to which an instance of this class belongs.	Identifier	M	Party ID
Industry Scheme	—	Specifies the sector classification scheme (or category) to which the party being classified is exposed or belongs.	Look-up	M	Industry Scheme
Industry Scheme Value	—	Specifies the specific sector classification (element) for a given sector classification scheme to which the party being classified is exposed or belongs.	Look-up	M	Industry Scheme Value
Industry Exposure	—	—	Number	O	—

SQL Table Generation Code

```
drop table if exists “PARTY INDUSTRY CLASSIFICATION”;
create table if not exists “PARTY INDUSTRY CLASSIFICATION”
(
    PARTY_INDUSTRY_ID        int             not null,
    PARTY_ID                 int             not null,
    INDUSTRY_SCHEME          varchar(10)     not null,
    INDUSTRY_SCHEME_VALUE    varchar(20)     not null,
    INDUSTRY_EXPOSURE        numeric(10,6)   not null,
    primary key (PARTY_INDUSTRY_ID),
);
```

A.3.5 Party Rating

The *Party Rating* represents the various proprietary and public ratings assigned to parties over time. This includes credit ratings for parties as well as more generic risk and performance ratings.

Table A.22: Attribute definitions of the *Party Rating* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Party Rating ID	PK	Unique system internal identifier for this class.	Identifier	M	—
Party ID	FK	System identifier for the party to which an instance of this class belongs.	Identifier	M	Party ID
Rating Scheme	—	Specifies the scheme name to which the given rating value belongs. The scheme name can be for instance S&P short term, S&P Long-term, Moody's Short Term, Moody's Long Term, etc.	Identifier	M	Rating Scheme
Rating Scheme Value	—	Stores the actual rating value given to the party.	String	M	—
Is Valid	—	Indicates whether an instance of this class is still currently valid or active. This flag has special significance in candidate and compound records.	Indicator	M	—
Valid From Date	—	Date from which the rating is valid.	Date	M	—
Valid To Date	—	Date on which this rating is no longer valid.	Date	O	—

SQL Table Generation Code

```
drop table if exists "PARTY RATING";
create table if not exists "PARTY RATING"
(
    PARTY_RATING_ID          int              not null,
    PARTY_ID                 int              not null,
    RATING_SCHEME            varchar(10)      not null,
    RATING_SCHEME_VALUE      varchar(20)      not null,
    IS_VALID                 tinyint(1)       not null,
    VALID_FROM_DATE          datetime         not null,
    VALID_TO_DATE            datetime,
    primary key (PARTY_RATING_ID),
);
```

A.4 The Role Model

A.4.1 Role Instrument Issuer

The *Role Instrument Issuer* is used to represent the role of a party as issuer for an instrument. An issuer is a party who creates an instrument and at least in the first instance is responsible and liable for any agreed payoff from an instrument throughout its life.

Table A.23: Attribute definitions of the *Role Instrument Issuer* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Issuer Role ID	PK	Unique system internal identifier for any instance of the Issuer class.	Identifier	M	–
Instrument ID	FK	Specifies the financial instrument for which the role is carried out.	Identifier	M	Instrument ID
Party ID	FK	Specifies the party who performs the specified role.	Identifier	M	Party ID

SQL Table Generation Code

```
drop table if exists "ROLE INSTRUMENT ISSUER";
create table if not exists "ROLE INSTRUMENT ISSUER"
(
    ISSUER_ROLE_ID          int          not null,
    INSTRUMENT_ID           int          not null,
    PARTY_ID                int          not null,
    primary key (INSTRUMENT_ID),
    unique (),
    unique (),
    ,
);
```

A.5 The Market Model

The Market Model consists of two attribute classes, *Instrument Price* and *Instrument Listing*, directly linked to the *Instrument Core* class of the Instrument model. It allows the representation of any market or estimated prices with a comprehensive set of descriptive attributes to distinguish different types of prices.

A.5.1 Instrument Price

The *Instrument Price* stores the prices for financial instruments. This includes instrument prices in the narrow conventional sense, as unit asset prices for instance used with equities, or percentage asset prices as used commonly for debt instruments, and for prices in a wider sense such as index values, prices expressed as yields, and others.

Table A.24: Attribute definitions of the *Instrument Price* class

Attribute Name	PK/FK	Description	Type	M/O	Domain
Price ID	PK	Unique system internal identifier for the price record.	Identifier	M	(Automatic)
Instrument ID	FK	Specifies the financial instrument for which this price applies.	Identifier	M	Instrument ID
Price Date	—	Specifies the date for which this price information is provided or to which it relates.	Date	M	—
Price Type	—	Specifies the price type; e.g., Daily Open, Daily Close, Daily Highest, Daily Lowest (or Intra-day) BID, ASK, PAID CALC, NAV.	Identifier	M	Price Type
Price Value	—	Specifies the price for this financial instrument after any adjustments.	Amount	M	—
Price Quotation Basis	—	Specifies the type of price quotation (e.g., percentage of nominal value).	Identifier	M	Quotation Basis
Price Currency	—	Specifies the currency used for this price.	Identifier	M	Currency (ISO 4217)
Price Lot Size	—	Specifies whether there is a preset and required trading size for the market where the instrument is traded. For example sometimes securities are traded only per 1000 units; e.g., in lots of 1000.	Amount	O	—

Continued

Table A.24: Continued

Attribute Name	PK/FK	Description	Type	M/O	Domain
Price Income Inclusion	—	Indicates whether the price is including/excluding entitlement to dividend, interest, or rights. Indicates whether the price value contains the entitlement ("Dirty"), or not ("Clean").	Identifier	O	Price Income Inclusion
Traded Volume	—	Specifies the volume that the instrument traded in this price. Volume traded on the day and market to which Price Date and Price Source Party ID refer.	Amount	O	—

SQL Table Generation Code

```
drop table if exists "INSTRUMENT PRICE";
create table if not exists "INSTRUMENT PRICE"
(
    PRICE_ID                    int              not null,
    INSTRUMENT_ID               int,
    PRICE_DATE                  datetime         not null,
    PRICE_TYPE                  varchar(10)      not null,
    PRICE_VALUE                 numeric(10,6)    not null,
    PRICE_QUOTATION_BASIS       varchar(10)      not null,
    PRICE_CURRENCY              varchar(10)      not null,
    PRICE_LOT_SIZE              numeric(10,4),
    PRICE_INCOME_INCLUSION      varchar(10),
    TRADED_VOLUME               numeric(10,4),
    primary key (PRICE_ID),
    unique (),
);
```

APPENDIX B

Code Lists

B.1 Analytic Scheme

Relevant Analytic Schemes depend on your application.

Table B.1: ANALYTIC SCHEME code list

Code	Code Name	Description
PRIC	Pricing Analytics	Analytics relating to the valuation of financial instruments.
RISK	Risk Analytics	Analytics relating to the risk measurement of portfolios.
PERF	Performance Analytics	Analytics relating to the performance measurement of portfolios.

B.2 Analytic Scheme Element

The list of exact Analytical Scheme Elements depends on your application.

Table B.2: ANALYTIC SCHEME ELEMENT code list

Analytic Scheme	Code	Code Name	Description
PRIC		Instrument Delta	Value calculated for an instrument.
PRIC		Instrument Delta	Value calculated for an instrument.
PRIC		Instrument Gamma	Value calculated for an instrument.
PRIC		CAPM Price	Value calculated for an instrument.
RISK		Value at Risk	Value calculated for a portfolio.
PERF		Jensen's Alpha	Value calculated for a portfolio.
...	...	...	...

B.3 Asset Classification

The asset classification can be arbitrarily detailed. The problem usually lies in finding the right classification aspects. This also tends to be counteracted by often inconsistent market language. For the purpose of this book, we chose to work with a very high level, but clear-cut classification.

Table B.3: ASSET CLASSIFICATION code list

Code	Code Name	Description
C	Commodity	Oil, Gold, Aluminium, Steel, Copper, Wheat, Coffee, etc.
D	Debt	Bonds, Floating Rate Notes, Treasury Bills, Asset Backed Securities, etc.
E	Equity	Ordinary Share/Common Stock, Preference Share, Depository Receipt.
F	Fund	Investment trust, Mutual Fund, Exchange Traded Fund, SICAV, REIT.
H	Structured Product	Exchangeable or Convertible Bond, Bond with Warrant attached, ...
R	Derivative Product	Option, Future, Forward, Swap, Warrant.
P	Portfolio	Custom Portfolio.
M	Market Interest Rate	Euribor, EONIA, LIBOR, U.S. Treasury Rate.
I	Index	Custom or market Index.
X	Cross Currency Rate	Exchange rate.

B.4 Cash Flow Element Type

Table B.4: CASH FLOW ELEMENT TYPE code list

Code	Code Name	Description
CC	Conversion Composite	Mandatory Conversion defined as the product or sum of its child elements
CF	Conversion Fixed	Mandatory Conversion of the remaining principal balance of an instrument into another instrument in one or more slices of a predetermined size where the size of each conversion slice is either the same or where the difference between payments is a fixed amount and/or a given proportion (fixed factor by which to multiply)
CV	Conversion Variable	Mandatory Conversion of the remaining principal balance of an instrument into another instrument in one or more slices the size of which is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...)

Table B.4: Continued

Code	Code Name	Description
DIF	Debt Income Fixed	Debt Style Income (Coupon) in one or more payments of a predetermined size where the size of each payment is either the same or where the difference between payments is a fixed amount and/or a given proportion (fixed factor by which to multiply). The size of the principal on which the calculated dividend payment is based is the sum of the remaining balance payable from all redemption elements of the instrument.
DIV	Debt Income Variable	Debt Style Income in one or more payments the size of which is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...). The size of the principal on which the calculated dividend payment is based is the sum of the remaining balance payable from all redemption elements of the instrument.
DRF	Debt Redemption Fixed	Debt Style Redemption in one or more payments of a predetermined size where the size of each payment is either the same or where the difference between payments is a fixed amount and/or a given proportion (fixed factor by which to multiply)
DRV	Debt Redemption Variable	Debt Style Redemption in one or more payments the size of which is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...)
EIF	Equity Income Fixed	(Preference Shares) Equity Style Dividend in one or more payments of a predetermined size where the size of each payment is either the same or where the difference between payments is a fixed amount and/or a given proportion (fixed factor by which to multiply). The size of the principal on which the calculated dividend payment is based is the sum of the remaining balance payable from all redemption elements of the instrument.
EIV	Equity Income Variable	Equity Style Dividend in one or more payments the size of which is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...). The size of the principal on which the calculated dividend payment is based is the sum of the remaining balance payable from all redemption elements of the instrument.

Continued

Table B.4: Continued

Code	Code Name	Description
ERF	Equity Redemption Fixed	(Preference Shares) Equity Style Redemption in one or more payments of a predetermined size where the size of each payment is either the same or where the difference between payments is a fixed amount and/or a given proportion (fixed factor by which to multiply).
ERV	Equity Redemption Variable	(Funds) Equity Style Redemption in one or more payments the size of which is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...).
EX	Equity Discretionary Dividend and Redemption	Combined Discretionary Equity Dividend and Redemption. Equity Style Dividend where the size of each payment is entirely discretionary at any point in time and constrained by the residual accumulated distributable income available at any point in time. This is the standard case for equities. Equity Style Redemption where the size of the redemption payment is discretionary at any point in time and constrained by the residual accumulated net realisable asset value that can be distributed available at any point in time. This is the standard case for equities.
FF	Credit Default Protection—Fixed Payout	Credit Default Protection that pays the holder a fixed amount or fixed proportion of the principal at risk for the first qualifying credit event related to each reference entity if the event falls within the exposure range of the tranche defined for the cash flow element where applicable.
FV	Credit Default Protection—Variable Payout	(Digital) Option that pays the holder an amount related to an underlying reference quantity (such as the instruments own market price after the event or some other reference quantity) for the first qualifying credit event related to each reference entity if the event falls within the exposure range of the tranche defined for the cash flow element where applicable.
OAC	Option Average Call	Asian one-off Call Option on a given underlying.
OAP	Option Average Put	Asian one-off Put Option on a given underlying.
OASFC	Option Average Series Fixed Call	Series of Asian Call Options each with a predetermined strike price set a certain amount and/or proportion above or below the preceding one.
OASFP	Option Average Series Fixed Put	Series of Asian Put Options each with a predetermined strike price set a certain amount and/or proportion above or below the preceding one.

Table B.4: Continued

Code	Code Name	Description
OASVC	Option Average Series Variable Call	Series of Asian Call Options each with a strike price set a certain amount or proportion above or below the average price of the underlying.
OASVP	Option Average Series Variable Put	Series of Asian Put Options each with a strike price set a certain amount or proportion above or below the average price of the underlying.
OCCF	Option Conversion Call Fixed	Option for the issuer to ask for conversion of the remaining principal balance, or a fraction of it defined by a child element, into another instrument on one or several dates where the proportion of the target instrument received by the investor in relation to any given amount of principal is a fixed amount and/or a given proportion (fixed factor by which to multiply).
OCCV	Option Conversion Call Variable	Option for the issuer to ask for conversion of the remaining principal balance, or a fraction of it defined by a child element, into another instrument on one or several dates where the proportion of the target instrument received by the investor in relation to any given amount of principal is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...).
OCPF	Option Conversion Put Fixed	Option for the investor to ask for conversion of the remaining principal balance, or a fraction of it defined by a child element, into another instrument on one or several dates where the proportion of the target instrument received by the investor in relation to any given amount of principal is either the same or where the difference between payments is a fixed amount and/or a given proportion (fixed factor by which to multiply).
OCPV	Option Conversion Put Variable	Option for the investor to ask for conversion of the remaining principal balance, or a fraction of it defined by a child element, into another instrument on one or several dates where the proportion of the target instrument received by the investor in relation to any given amount of principal is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...).
ODC	Option Digital Call	Digital one-off Call Option on a given underlying.
ODP	Option Digital Put	Digital one-off Put Option on a given underlying.

Continued

Table B.4: Continued

Code	Code Name	Description
ODSFC	Option Digital Series Fixed Call	Series of Digital Call Options each with a predetermined strike price set a certain amount and/or proportion above or below the preceding one.
ODSFP	Option Digital Series Fixed Put	Series of Digital Put Options each with a predetermined strike price set a certain amount and/or proportion above or below the preceding one.
ODSVC	Option Digital Series Variable Call	Series of Digital Call Options each with a strike price set a certain amount or proportion above or below the average price of the underlying.
ODSVP	Option Digital Series Variable Put	Series of Digital Put Options each with a strike price set a certain amount or proportion above or below the average price of the underlying.
OIC	Option Barrier Knock-in Call	Barrier Knock-in one-off Call Option on a given underlying.
OIP	Option Barrier Knock-in Put	Barrier Knock-in one-off Put Option on a given underlying.
OISFC	Option Barrier Knock-in Series Fixed Call	Series of Barrier Knock-in Call Options each with a predetermined knock-in price and strike price set a certain amount and/or proportion above or below the preceding one.
OISFP	Option Barrier Knock-in Series Fixed Put	Series of Barrier Knock-in Put Options each with a predetermined knock-in price and strike price set a certain amount and/or proportion above or below the preceding one.
OISVC	Option Barrier Knock-in Series Variable Call	Series of Barrier Knock-in Call Options each with a knock-in price and strike price set a certain amount or proportion above or below the average price of the underlying.
OISVP	Option Barrier Knock-in Series Variable Put	Series of Barrier Knock-in Put Options each with a knock-in price and strike price set a certain amount or proportion above or below the average price of the underlying.
OLC	Option Look-back Call	Look-back one-off Call Option on a given underlying.
OLP	Option Look-back Put	Look-back one-off Put Option on a given underlying.
OOC	Option Ordinary Call	Ordinary one-off Call Option on a given underlying.
OOP	Option Ordinary Put	Ordinary one-off Put Option on a given underlying.
OOSFC	Option Ordinary Series Fixed Call	Series of Ordinary Call Options each with a predetermined strike price set a certain amount and/or proportion above or below the preceding one.
OOSFP	Option Ordinary Series Fixed Put	Series of Ordinary Put Options each with a predetermined strike price set a certain amount and/or proportion above or below the preceding one.

Table B.4: Continued

Code	Code Name	Description
OOSVC	Option Ordinary Series Call	Series of Ordinary Call Options each with a strike price set a certain amount or proportion above or below the price of the underlying.
OOSVP	Option Ordinary Series Put	Series of Ordinary Put Options each with a strike price set a certain amount or proportion above or below the price of the underlying.
ORCF	Option Redemption Call Fixed	Option for the issuer to redeem the remaining principal balance, or a fraction of it defined by a child element, on one or several dates where the proportion of cash received by the investor in relation to any given amount of principal is either the same or where the difference between payments is a fixed amount and/or a given proportion (fixed factor by which to multiply).
ORCV	Option Redemption Call Variable	Option for the issuer to redeem the remaining principal balance, or a fraction of it defined by a child element, on one or several dates where the proportion of cash received by the investor in relation to any given amount of principal is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...).
ORPF	Option Redemption Put Fixed	Option for the investor to ask for redemption of the remaining principal balance, or a fraction of it defined by a child element, on one or several dates where the proportion of cash received by the investor in relation to any given amount of principal is either the same or where the difference between payments is a fixed amount and/or a given proportion (fixed factor by which to multiply).
ORPV	Option Redemption Put Variable	Option for the investor to ask for redemption of the remaining principal balance, or a fraction of it defined by a child element, on one or several dates where the proportion of cash received by the investor in relation to any given amount of principal is related to an underlying reference quantity (Instrument, Price, Interest Rate, Volatility, Index, ...).
OS	Option Special	Special option cash flow element.
OSC	Option Special Chooser	Chooser Option on a given underlying.
OSM	Option Special Composite	Composite Option defined as the product or sum of its child elements.
OSX	Option Special Extensible	Extensible Option on a given underlying.

Continued

Table B.4: Continued

Code	Code Name	Description
OXC	Option Barrier Knock-out Call	Barrier Knock-out one-off Call Option on a given underlying.
OXP	Option Barrier Knock-out Put	Barrier Knock-out one-off Put Option on a given underlying.
OXSFC	Option Barrier Knock-out Series Fixed Call	Series of Barrier Knock-out Call Options each with a predetermined Knock-Out price and strike price set a certain amount and/or proportion above or below the preceding one.
OXSFC	Option Look-back Series Call	Series of Look-back Call Options each for a predetermined time period.
OXSFP	Option Barrier Knock-out Series Fixed Put	Series of Barrier Knock-out Put Options each with a predetermined Knock-out price and strike price set a certain amount and/or proportion above or below the preceding one.
OXSFP	Option Look-back Series Put	Series of Look-back Put Options each for a predetermined time period.
OXSVC	Option Barrier Knock-out Series Variable Call	Series of Barrier Knock-out Call Options each with a Knock-out price and strike price set a certain amount or proportion above or below the average price of the underlying.
OXSVP	Option Barrier Knock-out Series Variable Put	Series of Barrier Knock-out Put Options each with a Knock-out price and strike price set a certain amount or proportion above or below the average price of the underlying.

B.5 Cash Flow Fixing Type

Table B.5: CASH FLOW FIXING TYPE code list

Code	Code Name	Description
ACR	Accrued Value	
AAC	Actual Accrued Value	Balance of Accrued Value from option series and similar accruing cashflows
ADV	Actual Dividend Rate	Actual Dividend Rate for next dividend
APF	Actual Pool Factor	Current Value of Pool Factor
ABL	Actual Principal Balance	Current Value of Remaining Principal Balance
ARD	Actual Redemption Rate	Actual Redemption Rate for next redemption
ASP	Actual Strike Price	Strike Price for next option in a series
CNP	Conversion Premium	
CNR	Conversion Rate	

Table B.5: Continued

Code	Code Name	Description
CPN	Coupon Rate	
DIV	Dividend Rate	
POL	Pool Factor	
PRI	Principal Balance	
RED	Redemption Rate	
STR	Strike Price	
UDX	Underlying Index Rate	Current Value of Underlying Index Rate
UIN	Underlying Interest Rate	Current Value of Underlying Interest Rate
UVL	Underlying Volatility	Current Value of Volatility of Underlying

B.6 Cash Flow Schedule Type

Table B.6: Cash Flow Schedule Type code list

Code	Code Name	Description
ACC	Accrual Schedule	Schedule to specify the start of accrual period(s) unless first accrual start on date given in Period Begin attribute on cash flow element.
DIV	Dividend Schedule	Schedule to specify the value dates for cash flow payments such as dividends.
SET	Settlement Schedule	Schedule to specify the dates for settlement of cash flow payments.
FIX	Fixing Schedule	Schedule to specify the dates for fixings to be made.
CRE	Credit Event Schedule	Schedule to specify the credit event types applicable to a credit protection cash flow element.
OBS	Observation Schedule for Fixings	Schedule to specify the dates for on which observations for fixings should be made.
RES	Reset Schedule for Stepped Cash flows	Schedule to specify the dates on which the steps for stepped cash flows can be adjusted unless this can be done on each fixing. This can be a single instance specifying a single reset or a regular series or a several instances specifying several irregular resets.
EXR	Simple Exercise Schedule for one off options	Schedule to specify the begin and end of the exercise period of an option unless this is already given by the Period start and End dates on the cash flow element.

Continued

Table B.6: Continued

Code	Code Name	Description
EXS	Exercise Period Start Date Schedule	Schedule to specify the begin of the exercise periods for an option series unless this is given by the Period Begin attribute on cash flow element or is the same as the exercise period end.
EXE	Exercise Period End Date Schedule	Schedule to specify the beginning of the exercise periods for an option series.
LBK	Simple Look back Schedule for one off look back periods	Schedule to specify the begin and end of the look back period for look back and averaging cash flows.
LBS	Look back Period Start Date Schedule	Schedule to specify the begin of look back periods for serial look back and averaging cash flows.
LBE	Look back Period End Date Schedule	Schedule to specify the end of look back periods for serial look back and averaging cash flows.

B.7 Constituent Function

Describes the function, a record on a constituent has in a portfolio of instruments.

Table B.7: Constituent Function code list

Code	Code Name
POR	Position Ordered
MOR	Movement in Position Ordered
PPL	Position Placed
MPL	Movement in Position Placed
PEX	Position Executed
MEX	Movement in Position Executed
PAL	Position Allocated
MAL	Movement in Position Allocated
PSE	Position Settled
MSE	Movement in Position Settled
PRE	Position Restricted
MRE	Movement in Position Restricted
BAL	Snapshot Balance

B.8 Currency (ISO 4217)

See http://www.iso.org/iso/support/currency_codes_list-1.htm for the full official list of codes.

Table B.8: CURRENCY (ISO 4217) code list

Code	Code Name	Country / Entity
ARS	Argentine Peso	Argentina
AUD	Australian Dollar	Australia
BGN	Bulgarian Lev	Bulgaria
BRL	Brazilian Real	Brazil
CAD	Canadian Dollar	Canada
CHF	Swiss Franc	Switzerland
CNY	Chinese Yuan Renminbi	China
CZK	Czech Koruna	Czech Republic
DKK	Danish Krone	Denmark
EEK	Estonian Kroon	Estonia
EUR	Euro	Austria, Belgium, Cyprus, Finland, France, Germany, Greece, Ireland, Italy, Luxembourg, Malta, Netherlands, Portugal, Slovakia, Slovenia, Spain
GBP	Pound Sterling	United Kingdom
HKD	Hong Kong Dollar	Hong Kong (China)
HRK	Croatian Kuna	Croatia
HUF	Hungarian Forint	Hungary
IDR	Indonesian Rupiah	Indonesia
ISK	Icelandic Krona	Iceland
JPY	Japanese Yen	Japan
KRW	South Korean Won	South Korea
LTL	Lithuanian Litas	Lithuania
LVL	Latvian Lats	Latvia
MYR	Malaysian Ringgit	Malaysia
NOK	Norwegian Krone	Norway
NZD	New Zealand Dollar	New Zealand
PHP	Philippine Peso	Philippines
PLN	Polish Zloty	Poland
RON	New Romanian Leu	Romania
RUB	Russian Rouble	Russian Federation
SEK	Swedish Krona	Sweden
SGD	Singapore Dollar	Singapore
SKK	Slovak Koruna	Slovakia
THB	Thai Baht	Thailand

Continued

Table B.8: Continued

Code	Code Name	Country / Entity
TRY	New Turkish Lira	Turkey
TWD	New Taiwan Dollar	Taiwan
USD	US Dollar	United States of America
ZAR	South African Rand	South Africa

B.9 Date Offset Rule

Table B.9: Date Offset Rule code list

Code	Code Name
T+0C	Settlement (Fixing) takes place in the same date as Trading (Record)
T+1C	Settlement takes place 1 calendar day after Trading date
T+1B	Settlement takes place 1 business day after Trading date
T-1C	Instr. goes Ex entitlement 1 calendar day before Record date
T-1B	Instr. goes Ex entitlement 1 business day before Record date
T+2C	Settlement takes place 2 calendar days after Trading date
T+2B	Settlement takes place 2 business days after Trading date
T-2C	Instr. goes Ex entitlement 2 calendar days before Record date
T-2B	Instr. goes Ex entitlement 2 business days before Record date
T+3C	Settlement takes place 3 calendar days after Trading date
T+3B	Settlement takes place 3 business days after Trading date
T-3C	Instr. goes Ex entitlement 3 calendar days before Record date
T-3B	Instr. goes Ex entitlement 3 business days before Record date
T+4C	Settlement takes place 4 calendar days after Trading date
T+4B	Settlement takes place 4 business days after Trading date
T-4C	Instr. goes Ex entitlement 4 calendar days before Record date
T-4B	Instr. goes Ex entitlement 4 business days before Record date
T-5B	Instr. goes Ex entitlement 5 business days before Record date
T-5C	Instr. goes Ex entitlement 5 calendar days before Record date
T-6B	Instr. goes Ex entitlement 6 business days before Record date
T-6C	Instr. goes Ex entitlement 6 calendar days before Record date
T-7B	Instr. goes Ex entitlement 7 business days before Record date
T-7C	Instr. goes Ex entitlement 7 calendar days before Record date
T-8C	Instr. goes Ex entitlement 8 calendar days before Record date
T-9C	Instr. goes Ex entitlement 9 calendar days before Record date
T-10C	Instr. goes Ex entitlement 10 calendar days before Record date
T-11C	Instr. goes Ex entitlement 11 calendar days before Record date
T-12C	Instr. goes Ex entitlement 12 calendar days before Record date
T-13C	Instr. goes Ex entitlement 13 calendar days before Record date

Table B.9: Continued

Code	Code Name
T-14C	Instr. goes Ex entitlement 14 calendar days before Record date
T-15C	Instr. goes Ex entitlement 15 calendar days before Record date
T-16C	Instr. goes Ex entitlement 16 calendar days before Record date
T-17C	Instr. goes Ex entitlement 17 calendar days before Record date
T-18C	Instr. goes Ex entitlement 18 calendar days before Record date
T-19C	Instr. goes Ex entitlement 19 calendar days before Record date
T-20C	Instr. goes Ex entitlement 20 calendar days before Record date

B.10 Date Roll Rule

Table B.10: Date Roll Rule code list

Code	Code Name
NOC	Don't move the date.
FOL	Move date to next business day.
PBD	Move the date to the previous business day.
PCD	Move date to the previous calendar date.
EMF	Move the date to the first business day after end of the previous month.
EMP	Move date to the end date of the previous month.
FLI	Move date to the next business date in the same month. If the next business date is in the following month don't move (ISMA).
FLC	Move date to next calendar date.
IMD	This convention calculates the third Wednesday of the calendar month.
MFL	Modified Following: Move the date to the next business day unless this is in a new months.
MPR	If the date is a business day, don't move it, otherwise move the date to the previous business day.

B.11 Day Count Convention

Table B.11: Day Count Convention code list

Code	Code Name
ACT_ACT	Actual / Actual
ACT_UST	Actual / Actual (US Treasury)
ACT_UST_PROPER	US Actual—second variant
ACT_OAT	Actual / Actual (French OAT)

Continued

Table B.11: Continued

Code	Code Name
ACT_365	Actual / 365
ACT_365E	Actual / 365 Even
ACT_365U	Actual / 365 Uneven
ACT_365_JPY	Actual / 365 Japanese variant
ACT_365_ISDA	Actual / 365 ISDA
END_365	Actual / 365 with modifications for accrued interest year fractions
ACT_360	Actual / 360
_30_360	30 / 360
_360E_ISDA	30 / 360 ISDA
_30_360_PSA	30 / 360 PSA
_30_360_SIA	30 / 360 SIA
_30E_360	30E / 360 Eurobonds
_30_360_ITL	30 / 360 Italian variant
_30_360_CHF	30 / 360 Swiss variant
_30_360_USD	30 / 360 U.S. variant
END_ACC	Bullet Accrual at Maturity
ON_DEC	On declaration

B.12 Debit Credit

Table B.12: Debit Credit code list

Code	Code Name	Description
DR	Debit	
CR	Credit	

B.13 Income Event Type

Table B.13: Income Event Type code list

Code	Code Name
DVI	Interim Dividend
DVF	Final Dividend
DIV	Other Dividend
CPN	Coupon Payment
BUY	Buy-back Income Payment

B.14 Index Valuation Formula

Table B.14: INDEX VALUATION FORMULA code list

Code	Code Name
LAW	LASPEYRES WEIGHTED—Use current weights & factors in denominator.
PAW	PAASCHE—WEIGHTED—Factors in denominator.
LAV	LASPEYRES AVG—Use current weights but no factors.
PAV	PAASCHE AVG—Historical weights and no factors in denominator.

B.15 Index Valuation Variable

Table B.15: INDEX VALUATION VARIABLE code list

Code	Code Name
NSO	Number of Shares Outstanding (Equity)
MCA	Market Capitalisation (Equity)
PRI	Price (Debt, Equity)
NAO	Nominal Amount Outstanding (Debt)
MVA	Market Value of Amount Outstanding (Debt)
NYI	Nominal Yield (Debt)
MYL	Market Yield

B.16 Index Weighting Variable

Table B.16: INDEX WEIGHTING VARIABLE code list

Code	Code Name
NSO	Number of Shares Outstanding (Equity)
MCA	Market Capitalisation (Equity)
FRE	Free Float (Equity)
PRI	Price (Debt, Equity)
NAO	Nominal Amount Outstanding (Debt)
MVA	Market Value of Amount Outstanding (Debt)
NYI	Nominal Yield (Debt)
MYL	Market Yield

B.17 Industry Scheme

Table B.17: Industry Scheme code list

Code	Code Name
STE	ACI Short Term European Paper (STEP) classification
ICB	Dow Jones EURO STOXX industry classification
FTID	Financial Times Industry Classification
GICS	Global Industry Classification Standard
ISIC	International Standard Industrial Classification
MSC	Morgan Stanley Capital International (MSCI) classification
NACER11	NACE industry classification Rev 1.1
NACER20	NACE industry classification Rev 2
NAICS02	North American Industry Classification System 2002
NAICS07	North American Industry Classification System 2007
SIC	SIC industry classification
UKSIC03	United Kingdom Standard Industrial Classification of Economic Activities (2003)
UKSIC07	United Kingdom Standard Industrial Classification of Economic Activities (2007)

B.18 Industry Scheme Value

Usually a rather long list that is best looked up on the relevant web sites for the various industry schemes. In this book, we use the following, shortened industry classification.

Table B.18

Code	Code Name
02	Oil GAS Producers
04	Oil Equipment, Services Distribution
06	Chemicals
08	Forestry Paper
10	Industrial Metals
12	Mining
14	Construction Materials
16	Aerospace Defence
18	General Industrials
20	Electronic Electrical Equipment
22	Industrial Engineering
24	Industrial Transportation
26	Support Services
28	Automobiles Parts
30	Beverages
32	Food Producers
34	Household Goods

Table B.18: Continued

Code	Code Name
36	Leisure Goods
38	Personal Goods
40	Tobacco
42	Health Care Equipment Services
44	Pharmaceuticals Biotechnology
46	Food Drug Retailers
48	General Retailers
50	Media
52	Travel Leisure
54	Fixed Line Telecommunications
56	Mobile Telecommunications
58	Electricity
60	Gas, Water Multi-utilities
62	Banks
64	Non-life Insurance
66	Life Insurance
68	Real Estate
70	General Financial
72	Equity Investment Instruments
74	Non-equity Investment Instruments
76	Software Computer Services
78	Technology Hardware Equipment

B.19 Instrument Status

Table B.19: Instrument Status code list

Code	Code Name	Description
ANN	Announced	The instrument has been announced to the market but no concrete steps towards issuance have been made public yet.
ISS	Issued	The instrument has been successfully issued and is now actively traded/held by investors.
NEW	Un-issued	The instrument is about to be, but not yet issued.
PLN	Planned	The instrument is in the status of planning, no issuance process has been embarked upon yet.
RED	Redeemed	The instrument has been fully redeemed and is not active anymore.
REJ	Rejected	The planned instrument has been rejected by the relevant regulatory authority.
TER	Terminated	The instrument was terminated.

B.20 Issuance Date Type

Table B.20: Issuance Date Type code list

Code	Code Name	Description
ACCRUALSTART	Accrual Start Date	Also called Dated Date: Date on which coupon interest first begins to accrue on a debt security.
ALLOTMENT	Allotment Date	The date on which shares are allotted to applicants for the current issue transaction.
ANNOUNCEMENT	Announcement Date	The date on which the current issue transaction is announced to the market, generally includes the publication of the prospectus.
AUCTION	Auction Date	The date on which applicants for the bond issue can make bids for the security.
BOOKBUILDING	Book-building Start Date	For listed, publicly dated, quoted shares: for EQUITY only; a lower priority variable, relevant for detailed recording of transaction data for listed and quoted shares (to be available in Phase 2 for equity).
OFFERCLOSING	Closing Date	Date after which no bids for initial purchases of this issuance transactions are accepted from potential investors. In practise, this may often coincide with the clearing and settlement date and/or the payment date.
DEALING	Dealing Date	Date on which units/shares of a fund are issued.
ESTIMATION	Estimation Date	The Is Estimate attribute specifies if the data in this instance have been estimated.
FILING	Filing Date	Date on which the issuing company formally asked for the registration of the issuance (by presenting the required filing according to the law).
FIRSTTRADING	First Trading Date	The first date instruments from this issuance transactions can be traded on an exchange.
INDICATIVE	Indicative Issue Date	Indicative Issue Date. To be used in the absence of any specific information about the nature of the date.
ISINCREATION	ISIN Code Creation Date	Date on which the ISIN code was allocated to the security.
LAUNCH	Launch Date	The formal date in which the very first issue, or initial (public) offering is launched.
OFFER	Offer Date	Date on which the offer has been made to potential investors.

Table B.20: Continued

Code	Code Name	Description
PAYMENT	Payment Date	The date on which the underwriters of an issue of bonds must effect payment for their allotment. Also settlement date.
PREMARKETING	Pre-marketing Start Date	For listed, publicly dated, quoted shares: for EQUITY only; a lower priority variable, relevant for detailed recording of transaction data for listed and quoted shares (to be available in Phase 2 for equity).
PRICING	Pricing Date	The date on which the final price and terms are set, prior to a formal offering.
SETTLEMENT	Settlement Date	Clearing and settlement date: for both cleaning lines.

B.21 Issuance Transaction Type

Table B.21: Issuance Transaction Type code list

Code	Code Name	Description
IPO	Initial public offering (primary)	Applies to Shares (Equity and Funds); initial tranche being offered by the issuer to the general public.
IPS	Initial public offering (secondary)	Applies to Shares (Equity and Funds); initial tranche being offered by an investor, who received it via a non-public offering from the issuer, to the general public.
SPO	Secondary public offering (primary)	Shares; secondary tranche from issuer to market.
SPS	Secondary public offering (secondary)	Shares; secondary tranche from market participant to open market.
BBA	Buy back	Repurchase of outstanding shares/units or amounts by the issuing company. Results in a reduction in the number of shares/ amounts outstanding.
SSP	Stock split	Change in minimum denomination and nominal value. Usually 1 existing share is exchanged for x shares with the new nominal value.
DEN	Re-denomination	Change of the nominal currency of an instrument, e.g., from Deutsche Mark to Euro.

Continued

Table B.21: Continued

Code	Code Name	Description
PAY	Payment only	
RII	Rights issuance	If the market is really short and nobody wants to buy your shares, how to sell? Propose right to buy new shares at a later period. Price can be ex- (excluding) & cum (including) rights issued. Rights have their own ISIN and expire after a number of weeks.
MEI	Merge issuance	Issuance of shares due to a merger of two companies.
CON	Conversion of debt to shares	In case that debt cannot be served anymore, one option is to convert it into shares, leading to an increase in the number outstanding of the company.
ESO	Execution of Stock Options	
PRIVP	Private Placement	Indicates whether the bond is issued through a Private placement (Y/N).
RDEN	Redenomination	
STKS	Stock Split	

B.22 Numbering Scheme

The following list of possible Numbering Schemes is merely illustrative and holds no claim on completeness.

Table B.22: Numbering Scheme code list

Code	Code Name
BIC	Bank Identification Code (BIC)
BLZ	Bankleitzahl (DE)
BEI	BEI code
BLO	Bloomberg code
CEDL	Cedel code
CINS	CINS code
COMM	Common code
ISOCCY	Currency code (ISO 3166)
CUSP	CUSIP code
DNS	Dun & Bradstreet code
EUCL	Euroclear code

Table B.22: Continued

Code	Code Name
FTI	FTID code
ISIN	ISIN code
MIC	Market Identification Code (MIC)
RTRI	Reuters (RIC) code
SNP	S&P code
SEDOL	SEDOL code
TKF	Telekurs Financial code
THO	Thomson code
WMD	WMD code
...	...

B.23 Portfolio Version Scheme

Table B.23: Portfolio Version Scheme code list

Code	Code Name	Description
SNP	Snapshot View	The portfolio version represents a snapshot of the actual wealth represented by the portfolio.
JRN	Real Time Positions & Movements	The portfolio version contains real time positions and transactions.
HIS	Historical Balances	The portfolio version contains historical balances of the portfolio.

B.24 Price Income Inclusion

Table B.24: Price Income Inclusion code list

Code	Code Name	
A	Adjusted	The price is adjusted for upcoming income payments.
N	Not adjusted	The price is not adjusted for upcoming income payments.
P	Partially adjusted	The price is partially adjusted for upcoming income payments.
X	Not applicable	Not applicable.

B.25 Price Type

Table B.25: PRICE TYPE code list

Code	Code Name	Description
DOPBID	Daily Open Bid Price	Value of the daily bid price—at opening of market.
DOPASK	Daily Open Ask Price	Value of the daily ask price—at opening of market.
DOPMID	Daily Open Mid Price	Value of the daily mid price—at opening of market.
DOPPAY	Daily Open Paid Price	Value of the daily paid / transaction price—at opening of market.
DOPCLC	Daily Open Calculated Price	Value of the daily calculated or fair price—at opening of market.
DOPIND	Daily Open Indicative Price	Value of the daily indicative price (dealer quote)—at opening of market.
DOPNAV	Daily Open Net Asset Value	Value of the daily Net Asset Value Per Share price—at opening of market.
DCLBID	Daily Close Bid Price	Value of the daily bid price—at market closure.
DCLASK	Daily Close Ask Price	Value of the daily ask price—at market closure.
DCLMID	Daily Close Mid Price	Value of the daily mid price—at market closure.
DCLPAY	Daily Close Paid Price	Value of the daily paid / transaction price—at market closure.
DCLCLC	Daily Close Calculated Price	Value of the daily calculated or fair price—at market closure.
DCLIND	Daily Close Indicative Price	Value of the daily indicative price (dealer quote)—at market closure.
DCLNAV	Daily Close Net Asset Value	Value of the daily Net Asset Value Per Share price—at market closure.
DHIBID	Daily Highest Bid Price	Value of the daily bid price—daily highest value.
DHIASK	Daily Highest Ask Price	Value of the daily ask price—daily highest value.
DHIMID	Daily Highest Mid Price	Value of the daily mid price—daily highest value.
DHIPAY	Daily Highest Paid Price	Value of the daily paid / transaction price—daily highest value.
DHICLC	Daily Highest Calculated Price	Value of the daily calculated or fair price—daily highest value.
DHIIND	Daily Highest Indicative Price	Value of the daily indicative price (dealer quote)—daily highest value.
DHINAV	Daily Highest Net Asset Value	Value of the daily Net Asset Value Per Share price—daily highest value.
DLOBID	Daily Lowest Bid Price	Value of the daily bid price—daily lowest value.
DLOASK	Daily Lowest Ask Price	Value of the daily ask price—daily lowest value.
DLOMID	Daily Lowest Mid Price	Value of the daily mid price—daily lowest value.
DLOPAY	Daily Lowest Paid Price	Value of the daily paid / transaction price—daily lowest value.

Table B.25: Continued

Code	Code Name	Description
DLOCLC	Daily Lowest Calculated Price	Value of the daily calculated or fair price—daily lowest value.
DLOIND	Daily Lowest Indicative Price	Value of the daily indicative price (dealer quote)—daily highest value.
DLONAV	Daily Lowest Net Asset Value	Value of the daily Net Asset Value Per Share price—daily lowest value.
MAVBID	Monthly Average Bid Price	Monthly average of daily paid / transaction prices at market closure.
MAVMID	Monthly Average Ask Price	Monthly average of daily bid prices at market closure.
MAVASK	Monthly Average Mid Price	Monthly average of daily ask prices at market closure.
MAVPAY	Monthly Average Paid Price	Monthly average of daily Net Asset Values per share at market closure.
MAVCLC	Monthly Average Calculated Price	Minimum closing price registered by the data provider during the last 52 weeks.
MAVIND	Monthly Average Indicative Price	Maximum closing price registered by the data provider during the last 52 weeks.
MAVNAV	Monthly Average NAV Price	Average of transacted prices of the day weighted by the respective amount / number of securities transacted.
DVWAP	Daily Volume Weighted Average Price	Net asset value per share at the end of the day.
DBOOK	Book Value	
VOLAD	One Day Volatility	
VOLAW	One Week Volatility	
VOLAM	One Month Volatility	
VOLAY	One Year Volatility	
MINBID	Period Minimum Bid Price	
MINASK	Period Minimum Ask Price	
MINMID	Period Minimum Mid Price	
MINPAY	Period Minimum Paid Price	
MINCLC	Period Minimum Calculated Price	
MININD	Period Minimum Indicative Price	
MINNAV	Period Minimum Net Asset Value	
MAXBID	Period Maximum Bid Price	
MAXASK	Period Maximum Ask Price	

Continued

Table B.25: Continued

Code	Code Name	Description
MAXMID	Period Maximum Mid Price	
MAXPAY	Period Maximum Paid Price	
MAXCLC	Period Maximum Calculated Price	
MAXIND	Period Maximum Indicative Price	
MAXNAV	Period Maximum Net Asset Value	
AVGBID	Period Average Bid Price	
AVGASK	Period Average Ask Price	
AVGMID	Period Average Mid Price	
AVGPAY	Period Average Paid Price	
AVGCLC	Period Average Calculated Price	
AVGIND	Period Average Indicative Price	
AVGNAV	Period Average Net Asset Value	

B.26 Quotation Basis

Table B.26: Quotation Basis code list

Code	Code Name	Description
CCY	Currency per share/unit	Nominal Currency
PCL	Percentage of nominal—Clean	Percentage of par nominal value—excluding accrued interest
PDT	Percentage of nominal—Dirty	Percentage of par nominal value—including accrued interest
PNT	Points	Point (Indices)
UNT	Unit	Monetary amount per unit
PCT	Percentage	Percentage of nominal value
DDT	Discount—Dirty	
DCL	Discount—Clean	
DAC	Discount—Annualised Clean	
DAD	Discount—Annualised Dirty	
YLD	Yield	
YAN	Yield Annualised	

B.27 Party Account Amount Function

Table B.27: Party Account Amount Function code list

Code	Code Name
CAEV.ATAX	CAEV Additional Tax Amount
CAEV.CAPG	CAEV Capital Gains Amount
CAEV.CHAR	CAEV Charges/Fees Amount
CAEV.CINL	CAEV Cash in Lieu of Shares
CAEV.COUN	CAEV Country, National Federal Tax
CAEV.ENTL	CAEV Entitled Amount
CAEV.EUTR	CAEV EU Tax Retention Amount
CAEV.EXEC	CAEV Executing Broker's Amount
CAEV.FISC	CAEV Fiscal Stamp
CAEV.FLFR	CAEV Fully Franked Amount
CAEV.GRSS	CAEV Gross Amount
CAEV.INCE	CAEV Cash Incentive
CAEV.INDM	CAEV Indemnity Amount
CAEV.INTR	CAEV Interest Amount
CAEV.ISDI	CAEV Issue Discount Amount
CAEV.LADT	CAEV Local Tax (DE specific 1)
CAEV.LEVY	CAEV Payment Levy Tax
CAEV.LIDT	CAEV Local Tax (DE specific 2)
CAEV.LOCL	CAEV Local Tax
CAEV.LOCO	CAEV Local Broker's Commission
CAEV.LOTE	CAEV Local Tax (DE specific 3)
CAEV.LYDT	CAEV Local Tax (DE specific 4)
CAEV.MFDV	CAEV Manufactured Dividend
CAEV.MKTC	CAEV Market Claim
CAEV.NETT	CAEV Net Cash Amount
CAEV.OCMT	CAEV Original Currency and Ordered Amount
CAEV.PAMM	CAEV Paying/Sub-paying Agent Commission
CAEV.POST	CAEV Postage Amount
CAEV.PRIN	CAEV Principal or Corpus
CAEV.REDP	CAEV Redemption Premium Amount
CAEV.REGF	CAEV Regulatory Fees
CAEV.REIN	CAEV Reinvestment Amount
CAEV.RESU	CAEV Resulting Amount
CAEV.SHIP	CAEV Shipping Amount
CAEV.SOFE	CAEV Solicitation Fee
CAEV.SOIC	CAEV Sundry/Other Income Amount
CAEV.SPCN	CAEV Special Concessions
CAEV.STAM	CAEV Stamp Duty

Continued

Table B.27: Continued

Code	Code Name
CAEV.STEX	CAEV Stock Exchange Tax
CAEV.SUBS	CAEV Additional Subscription Costs
CAEV.TAXC	CAEV Amount of Tax Credit
CAEV.TAXR	CAEV Withholding Tax Amount
CAEV.TRAN	CAEV Transfer Tax
CAEV.TRAX	CAEV Transaction Tax
CAEV.TXDF	CAEV Tax Deferred Amount
CAEV.TXFR	CAEV Tax Free Amount
CAEV.TXRC	CAEV Reclaim of Taxes
CAEV.UNFR	CAEV Unfranked Amount
CAEV.VATA	CAEV Value-Added Tax
CAEV.WITF	CAEV Withholding of Foreign Tax
CAEV.WITL	CAEV Withholding of Local Tax
PFTX.BUYC	PTF TX Purchase Cost (Clean)
PFTX.BUYD	PTF TX Purchase Cost (Dirty)
PFTX.ACIP	PTF TX Accrued Interest Purchased
PFTX.ACIS	PTF TX Accrued Interest Sold
PFTX.ACIR	PTF TX Interest Accrual (Int. Rec)
PFTX.ACIO	PTF TX Interest Accrual (Int. Owing)
PFTX.SELC	PTF TX Sales Revenue (Clean)
PFTX.SELD	PTF TX Sales Revenue (Dirty)
PFTX.STMP	PTF TX Stamp Duty
PFTX.COMM	PTF TX Trade Commission
PFTX.IMRG	PTF TX Initial Margin
PFTX.VMRG	PTF TX Variation Margin
PFTX.TLEV	PTF TX Other Trade Charges/Taxes
PFTX.INVA	PTF TX Invoice Amount
PFTX.VATA	PTF TX VAT Amount
PFTX.TAXA	PTF TX Tax Amount
PFTX.PAYA	PTF TX Amount to be Paid
PFTX.RECA	PTF TX Amount to be Received
PFTX.CMV1	PTF TX Amount of Ledger Credit Movement 01
PFTX.CMV2	PTF TX Amount of Ledger Credit Movement 02
PFTX.CMV3	PTF TX Amount of Ledger Credit Movement 03
PFTX.CMV4	PTF TX Amount of Ledger Credit Movement 04
PFTX.CMV5	PTF TX Amount of Ledger Credit Movement 05
PFTX.CMV6	PTF TX Amount of Ledger Credit Movement 06
PFTX.CMV7	PTF TX Amount of Ledger Credit Movement 07
PFTX.CMV8	PTF TX Amount of Ledger Credit Movement 08
PFTX.CMV9	PTF TX Amount of Ledger Credit Movement 09

Table B.27: Continued

Code	Code Name
PFTX.CMV0	PTF TX Amount of Ledger Credit Movement 10
PFTX.DMV1	PTF TX Amount of Ledger Debit Movement 01
PFTX.DMV2	PTF TX Amount of Ledger Debit Movement 02
PFTX.DMV3	PTF TX Amount of Ledger Debit Movement 03
PFTX.DMV4	PTF TX Amount of Ledger Debit Movement 04
PFTX.DMV5	PTF TX Amount of Ledger Debit Movement 05
PFTX.DMV6	PTF TX Amount of Ledger Debit Movement 06
PFTX.DMV7	PTF TX Amount of Ledger Debit Movement 07
PFTX.DMV8	PTF TX Amount of Ledger Debit Movement 08
PFTX.DMV9	PTF TX Amount of Ledger Debit Movement 09
PFTX.DMV0	PTF TX Amount of Ledger Debit Movement 10

B.28 Party Account Scheme

Table B.28: Party Account Scheme code list

Code	Code Name
BLS	Balance Sheet
PNL	Profit and Loss Statement
CFL	Annual Cash Flow
OPS	Operative Data
PTF	Portfolio Accounts

B.29 Party Account Scheme Element

Table B.29: Party Account Scheme Element code list

Account Scheme	Code	Code Name
BLS	111100	Cash Balances
BLS	111200	Money Market Investments
BLS	112000	Other Debt due within 12 Months
BLS	121000	Stock & Work in Progress at Beginning of Period
BLS	122000	Stock & Work in Progress at End of Period
BLS	131000	Medium Term Financial Investments
BLS	132000	Long Term Financial Investments
BLS	133000	Debt due in more than 12 Months
BLS	141000	Fixed Assets other than Land

Continued

Table B.29: Continued

Account Scheme	Code	Code Name
BLS	142000	Accum. Depreciation on Fixed Assets other than Land
BLS	150000	Investments in Land
BLS	160000	Participations in other Companies
BLS	170000	Goodwill
BLS	210000	Creditor Claims due on Demand
BLS	220000	Creditor Claims due in 12 months or less
BLS	230000	Creditor Claims due in more than 12 months
BLS	240000	Reserves
BLS	250000	Accumulated Profit & Loss
BLS	261000	Share Capital—Nominal
BLS	261000	Share Premium
PNL	310000	Sales Revenue
PNL	320000	Cost of sales
PNL	330000	Gross Profit
PNL	410000	Administrative Salaries & Wages
PNL	420000	Marketing Costs
PNL	430000	Other Administrative Costs
PNL	440000	Depreciation Expenses
PNL	510000	Interest Received
PNL	520000	Interest Paid
PNL	610000	Profit before Interest & Taxes
PNL	620000	Profit before Taxes
PNL	630000	Taxes on Profit
PNL	640000	Profit after Interest & Taxes
PTF	110000EUR	Euro Cash Account
PTF	110000GBP	GBP Cash Account
PTF	110000USD	US Dollar Cash Account
PTF	110000JPY	Japanese Yen Cash Account
PTF	110000CHF	Swiss Franc Cash Account
PTF	110000PXC	Portfolio Cash Control Account
PTF	110000CXC	Client Cash Control Account
PTF	120000	Investment Positions at Cost
PTF	130000	Unrealised Investment Gains/Losses
PTF	140000	Realised Investment Gains/Losses
PTF	210000	Transaction Taxes
PTF	220000	Transaction Fees
PTF	230000	Management Fees
PTF	240000	Operating Fees
PTF	300000	Cash Inflows
PTF	400000	Cash Outflows

B.30 Party Account Status

Table B.30: Party Account Status code list

Code	Code Name	Description
O	Open	Account is open.
R	Reconciled	Account is reconciled.
C	Closed	Account is closed.
U	Un-posted	Account is not yet posted.

B.31 Party Type

Table B.31: Party Type code list

Code	Code Name
COM	Company or other commercial legal entity
GOV	Sovereign Government
AGY	Government Agency
MUN	Municipality
INT	International Organisation
BRA	Unincorporated Branch
DIV	Unincorporated Division
GRP	Unincorporated Group
IPR	Issuance program

B.32 Party Status

Table B.32: Party Status code list

Code	Code Name	Description
ACT	Active	The party is active and can operate without any legal impediment.
SUS	Suspended	
DIS	Dissolved	The party is dissolved.
ADM	In Administration	
ISV	Insolvent	The party is insolvent.
LIQ	In liquidation	The party is in liquidation.
LQT	Liquidated	The party has been liquidated.
TER	Terminated	The party is terminated by the relevant bodies.
REJ	Rejected	The creation of the party has been rejected by the relevant authorities.
UKN	Unknown	The status of the party is unknown.
PLN	Planned	The party is yet in the status of planning.

B.33 Rating Scheme

Table B.33: Rating Scheme code list

Code	Code Name
AMBFSTR	AM Best Financial Strength Rating
AMBLTDR	AM Best Long-Term Debt Rating
AMBLTIR	AM Best Long-Term Insurance Company Credit Rating
AMBLTNR	AM Best Long-Term Non-Insurance Company Credit Rating
AMBSTDR	AM Best Short-Term Debt Rating
AMBSTSR	AM Best Short-Term Issuer Credit Rating
BASELIIW	Basle II Weight
FERCLOSENDFND	Feri Closed-End Fund Rating
FERMUTFND	Feri Mutual Fund Rating
FITALTASSTMAN	Fitch Alternative Asset Managers Rating
FITASSTMAN	Fitch Asset Management Rating
FITCDOASSTMAN	Fitch CDO Asset Manager Rating
FITDISTRECOV	Fitch Distressed Recovery Rating
FITBANK	Fitch Individual Bank Rating
FITINSTLGT	Fitch Instrument Long-term Rating
FITINSTSHT	Fitch Instrument Short-term Rating
FITINVPROC	Fitch Investment Process Rating
FITLGTINSFSTRNG	Fitch Long-Term Insurer Financial Strength Rating
FITLGTISSDEF	Fitch Long-Term Issuer Default Rating
FITMFNDVOL	Fitch Managed Fund Volatility Rating
FITMFNDCRED	Fitch Managed Funds Credit Rating
FITREASSTMAN	Fitch Real Estate Asset Managers Rating
FITRECOV	Fitch Recovery Rating
FITSERV	Fitch Servicer Ratings
FITSHTINSFSTRNG	Fitch Short-Term Insurer Financial Strength Rating
FITSHTISSDEF	Fitch Short-Term Issuer Default Rating
FITSUPP	Fitch Support Rating
FITUSPUBFIN	Fitch U.S. Public Finance Rating
LGTPTYGEN	Long-term Generic Rating
LGTINSTGEN	Long-term Generic Rating
MDYBKFS	Moody's Bank Financial Strength Rating
MDYHYBK	Moody's Classification for Hybrid Baskets
MDYCDSR	Moody's Credit Default Swaps Rating
MDYEQFD	Moody's Equity Fund Rating
MDYFSTR	Moody's Financial Strength Rating
MDYHFQR	Moody's Hedge Fund Operations Quality Rating
MDYIMQR	Moody's Investment Manager Quality Rating
MDYLDSR	Moody's Lloyd's Syndicate Rating
MDYLDVR	Moody's Lloyd's Volatility Rating

Table B.33: Continued

Code	Code Name
MDYLTCO	Moody's Long-Term Corporate Obligation Rating
MDYLTIFS	Moody's Long-Term Insurance Financial Strength Rating
MDYLTIR	Moody's Long-Term Issuer Rating
MDYLGDA	Moody's Loss Given Default Assessment
MDYMKRS	Moody's Market Risk Rating
MDYMMBF	Moody's Money Market and Bond Fund Rating
MDYRECF	Moody's Real Estate Portfolio Cash Flow Volatility Rating
MDYSRVQR	Moody's Servicer Quality Rating
MDYSTCOR	Moody's Short-Term Corporate Obligation Rating
MDYSTIFSR	Moody's Short-Term Insurance Financial Strength Rating
MDYSTIR	Moody's Short-Term Issuer Rating
MDYSGLR	Moody's Speculative Grade Liquidity Rating
MDYSFIR	Moody's Structured Finance Issuer Rating
MDYFLTR	Moody's Structured Finance Long-Term Rating
MDYFSTIR	Moody's Structured Finance Short-Term Issuer Rating
MDYTQR	Moody's Trustee Quality Rating
MDYUMDOR	Moody's US Municipal Demand Obligation Rating
MDYUMLDR	Moody's US Municipal Long-Term Debt Rating
MDYUMSDR	Moody's US Municipal Short-Term Debt Rating
MSTARFR	Morningstar Fund Rating
SNPBANKFSR	S&P Bank Fundamental Strength Rating
SNPBANKSA	S&P Bank Survivability Assessment
SNPCPR	S&P Commercial Paper Rating
SNPFCQR	S&P Fund Credit Quality Rating
SNPFVR	S&P Fund Volatility Rating
SNPIFER	S&P Insurer Financial Enhancement Rating
SNPLIFSR	S&P Long-Term Insurer Financial Strength Rating
SNPSTIR	S&P Short-Term Issue Credit Rating
SNPLTIR	S&P Long-Term Issuer Credit Rating
SNPSMICR	S&P Short-Term Municipal Issue Credit Rating
SNPLMICR	S&P Long-Term Municipal Issuer Credit Rating
SNPPSFR	S&P Principal Stability Fund Rating
SNPR	S&P Recovery Rating
SNPSIFSR	S&P Short-Term Insurer Financial Strength Rating
SNPSICR	S&P Short-Term Issue Credit Rating
SNPLICR	S&P Long-Term Issuer Credit Rating
SNPSMICR	S&P Short-Term Municipal Issue Credit Rating
SNPLMICR	S&P Long-Term Municipal Issuer Credit Rating
SNPSFSE	S&P Structured Finance Servicer Evaluation
SNPSRR	S&P Swap Risk Rating

B.34 Repetition Period Type

Table B.34: REDEMPTION PERIOD TYPE code list

Code	Code Name
DY	Period in Days
WK	Period in Weeks
MO	Period in Months
YR	Period in Years
EV	Triggered by a Specific Event
DT	Single Specific Date

B.35 Unit

Table B.35: UNIT code list

Code	Code Name	Description
SHR	Shares/Units	
PRI	Principal	Type of percentage. Based in relative amount of what the principal is
CCY	Currency	
PNT	Points	
PCT	Percentage	Unit for truly percentage based instruments, like (H)CPI.
TOU	Troy ounces	
BAR	Barrels	
GAL	Gallons	
TON	Tonnes	
LBS	Pounds	
BSH	Bushels	
CEL	Celsius degrees	
MW	Megawatts	
INH	Inches	
HRS	Hours	

B.36 Underlying Type

Table B.36: Underlying Type code list

Code	Code Name
INS	Single Instrument
CMP	Complex underlying made up of several child elements
BSK	Basket of Reference Entities
PTF	Portfolio
IDX	Index
CRV	Curve

This page intentionally left blank

References

Arrow KJ. The role of securities in the optimal allocation of risk-bearing. *Review of Economic Studies*. 1953, reprint 1964;31:91–96.

Basel Committee on Banking Supervision. *Sound Practices for the Management and Supervision of Operational Risk*. Bank for International Settlements; 2003.

Black F, Scholes M. The pricing of options and corporate liabilities. *Journal of Political Economy*. 1973;81:637–659.

Brinson GP, Fachler N. Measuring non-us equity portfolio performance. *Journal of Portfolio Management*. Spring 1985;73–76.

Brinson GP, Hood R, Beebower GL. Determinants of portfolio performance. *Financial Analysts Journal*. 1986;42(Jul/Aug 4):39–48.

Chiang AC, Wainwright K. *Fundamental Methods of Mathematical Economics*. 4th ed. McGraw-Hill Professional; 2005.

Choudhry M. *Advanced Fixed Income Analysis*. Butterworth Heinemann; 2004.

Cochrane J. *Asset Pricing, revised edition*. University Presses of Cambridge; 2005.

Colin A. *Fixed Income Attribution*. New York: John Wiley & Sons Inc. Ltd; 2005.

Damodaran A. *Investment Valuation: Tools and Techniques for Determining the Value of Any Asset*. Wiley & Sons; 2002.

Date CJ. *Database in Depth: Relational Theory for Practitioners*. O'Reilly Media; 2005.

Date CJ. *The Relational Database Dictionary*, extended edition. Apress; 2008.

Debreu G. *Theory of Value*, Chapter 7. New York: John Wiley & Sons, Inc; 1959.

Dowd K. *An Introduction to Market Risk Measurement*. New York: John Wiley & Sons; 2002.

Duffie D. *Dynamic Asset Pricing Theory*. 3rd ed. Princeton University Press; 2001.

Greene WH. *Econometric Analysis*. 6th ed. Prentice Hall International; 2007.

Hansen LP, Richard SF. The role of conditioning information in deducing testable restrictions implied by dynamic asset pricing models. *Econometrica*. 1987;55:587–614.

Harrison JM, Kreps DM. Martingales and arbitrage in multiperiod securities markets. *Journal of Economic Theory*. 1979;20:381–408.

Huang CF, Litzenberger RH. *Foundations for Financial Economics*. Pearson; 1988.

Ingersoll JE. *Theory of Financial Decision Making*. Rowman & Littlefield; 1987.

Innmon, Zachman, Geiger. *Data Stores, Data Warehousing, and the Zachman Framework*, McGraw-Hill; 1997.

Mas-Collel A, Whinston MD, Green JR. *Microeconomic Theory*. Oxford University Press; 1995.

Merton RC. Theory of rational option pricing. *Bell Journal of Economics and Management Science*. 1973;4:141–183.

Nagatani K. On a theorem of arrow. *Review of Economic Studies*. 1975;42:483–485.

Nelson CR, Siegel AF. Parsimonious modeling of yield curves. *The Journal of Business*. 1987;60(4):473–489.

Radner R. Equilibrium under uncertainty. In: Arrow KJ, Intrilligator MD, eds. *Handbook of Mathematical Economics*, Vol. II. Chapter 20. North Holland, Inc; 1982.

Ross SA. A simple approach to the valuation of risky streams. *Journal of Business*. 1978;51:453–475.

Rubinstein M. The valuation of uncertain income streams and the price of options. *Bell Journal of Economics*. 1976;7:407–425.

Scholz H, Wilkens M. Interpreting Sharpe Ratios—The Market Climate Bias (July 20, 2006). Finance Letters, Forthcoming. Available at SSRN: http://ssrn.com/abstract=524842

Simsion G, Witt G. *Data Modeling Essentials*, 3rd ed. Morgan Kaufmann Series in Data Management Systems; 2004.

Varian HR. *Microeconomic Analysis*. New York: W.W. Norton & Company; 1992.

Varian HR. *Intermediate Microeconomics—A Modern Approach*. New York: W.W. Norton & Company; 2006.

Index

Page numbers followed by "f" indicate figures, "t" indicate tables, and "b" indicate boxes and formulas.