

1. Record Nr.	UNINA9910149552903321
Autore	Brambilla Marco
Titolo	Model-driven software engineering in practice // Marco Brambilla, Jordi Cabot, Manuel Wimmer
Pubbl/distr/stampa	San Rafael, Calif. (1537 Fourth Street, San Rafael, CA 94901 USA), : Morgan & Claypool, c2012
ISBN	1-60845-883-0
Descrizione fisica	1 online resource (184 p.)
Collana	Synthesis lectures on software engineering ; ; # 1
Altri autori (Persone)	CabotJordi WimmerManuel
Disciplina	005.1
Soggetti	Software engineering Computer programming - Computer simulation Model-integrated computing
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Part of: Synthesis digital library of engineering and computer science. Series from website.
Nota di bibliografia	Includes bibliographical references (p. 159-164).
Nota di contenuto	1. Introduction -- 1.1 Purpose and use of models -- 1.2 Modeling for software development -- 1.3 How to read this book -- 2. MDSE principles -- 2.1 MDSE basics -- 2.2 Lost in acronyms: the MD* jungle -- 2.3 Overview of the MDSE methodology -- 2.4 MDSE adoption in industry -- 2.5 Tool support -- 2.6 Criticisms of MDSE -- 3. MDSE use cases -- 3.1 Automating software development -- 3.2 System interoperability -- 3.3 Reverse engineering -- 4. Model-driven architecture (MDA) -- 4.1 MDA definitions and assumptions -- 4.2 The modeling levels: CIM, PIM, PSM -- 4.3 Mappings -- 4.4 General-purpose and domain-specific languages in MDA -- 4.5 Architecture-driven modernization -- 5. Integration of MDSE in your development process -- 5.1 Introducing MDSE in your software development process -- 5.2 Traditional development processes and MDSE -- 5.3 Agile and MDSE -- 5.4 Domain-driven design and MDSE -- 5.5 Test-driven development and MDSE -- 6. Modeling languages at a glance -- 6.1 Anatomy of modeling languages -- 6.2 General-purpose vs. domain-specific modeling languages -- 6.3 General-purpose modeling: the case of UML -- 6.4

UML extensibility: the middle way between GPL and DSL -- 6.5
Overview on DSLs -- 6.6 Defining modeling constraints (OCL) --
7. Developing your own modeling language -- 7.1 Metamodel-centric
language design -- 7.2 Example DSML: sWML -- 7.3 Abstract syntax
development -- 7.4 Concrete syntax development --
8. Model-to-model transformations -- 8.1 Model transformations and
their classification -- 8.2 Exogenous, out-place transformations -- 8.3
Endogenous, in-place transformations -- 8.4 Mastering model
transformations --
9. Model-to-text transformations -- 9.1 Basics of model-driven code
generation -- 9.2 Code generation through programming languages --
9.3 Code generation through M2T transformation languages -- 9.4
Mastering code generation -- 9.5 Excursus: code generation through
M2M transformations and TCS --
10. Managing models -- 10.1 Model interchange -- 10.2 Model
persistence -- 10.3 Model comparison -- 10.4 Model versioning --
10.5 Model co-evolution -- 10.6 Global model management -- 10.7
Model quality -- 10.7.1 Verifying models --
11. Summary -- Bibliography -- Authors' biographies.

Sommario/riassunto

This book discusses how model-based approaches can improve the daily practice of software professionals. This is known as Model-Driven Software Engineering (MDSE) or, simply, Model-Driven Engineering (MDE). MDSE practices have proved to increase efficiency and effectiveness in software development, as demonstrated by various quantitative and qualitative studies. MDSE adoption in the software industry is foreseen to grow exponentially in the near future, e.g., due to the convergence of software development and business analysis.
