LE LANGAGE C++
MASTER 1

Jean-Baptiste.Yunes@univ-paris-diderot.fr
U.F.R. d'Informatique
Université Paris Diderot - Paris 7

2013—2014

INFORMATIONS *ADMINISTRATIVES*

- Un Cours

`Jean-Baptiste.Yunes@univ-paris-diderot.fr`

  - Lundi 15h30—17h30
    - Salle 247E de la Halle aux Farines

- Des Travaux Pratiques
  - Lundi 8h30—10h30 (Isabelle Fagnot)
    - Salle 2031
  - Lundi 8h30—10h30 (Antoine Mamcarz)
    - Salle 2032

- Pour les Travaux Pratiques
  - Veiller à ce que vous possédiez un compte sur les machines de l'UFR (ce compte est différent de votre compte ENT)
  - Répartissez-vous également sur les deux groupes

La **notation**

- Un examen (en Janvier, date à préciser ultérieurement)
- Un projet (délivrance du sujet et date de soutenance à préciser ultérieurement)
- Pour être reçu :
  60% examen + 40% projet
- Si inférieur à 10, seconde session :
  60% session2 + 40% projet

Un **partiel**

- non obligatoire
- permet d'obtenir un *bonus* pour la note finale
- ne pas s'y rendre ou le rater ne pénalisera pas...
- on ne peut que gagner!

Instructions utiles :

consultez régulièrement l'entrée DIDEL du cours M LOV,

n'hésitez pas à utiliser les forums pour poser des questions,

marquez la page du cours dans vos signets

`http://www.liafa.univ-paris-diderot.fr/~yunes/cours/cpp/`

réfléchissez déjà à la composition de vos trinômes pour le projet à venir.

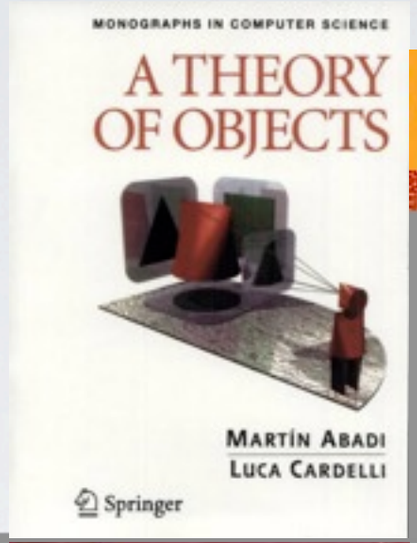
Éléments du cours :

1. Introduction
2. Un C plus plus ?
3. Le paradigme objet
4. Les classes
5. Du modèle aux classes
6. La spécialisation
7. La factorisation
8. L'héritage multiple
9. Les opérateurs
10. Les modèles
11. Les design patterns
12. Les exceptions
13. La Standard Template Library
14. La méta-programmation

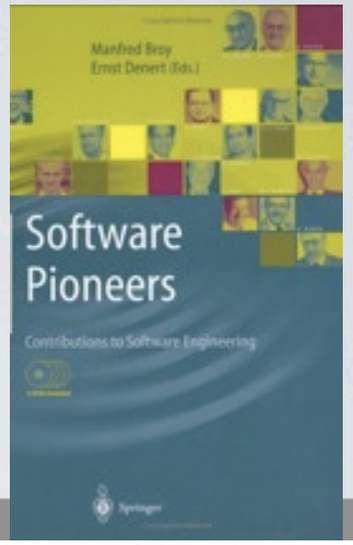
Quels sont les **buts** de ce cours ?

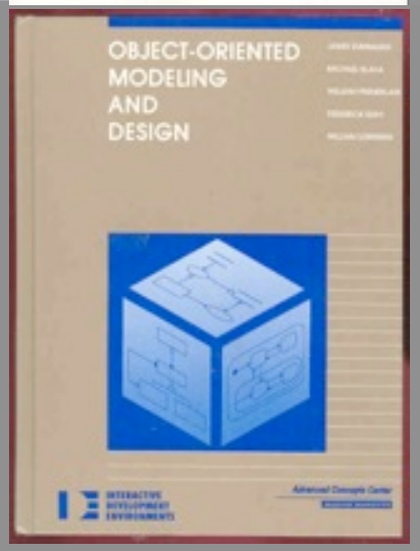
- Une connaissance de la problématique de l'analyse objet
- La maîtrise des éléments de base d'UML
- La connaissance de l'implémentation en C++ de fragments UML
- La maîtrise du langage C++ et de ses particularités

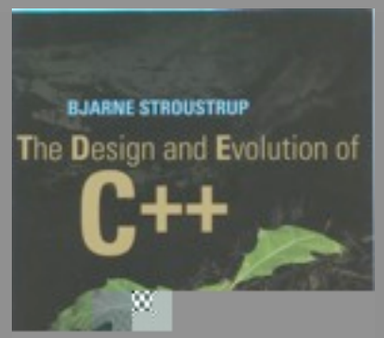
BIBLIOGRAPHIE


MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer

Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner

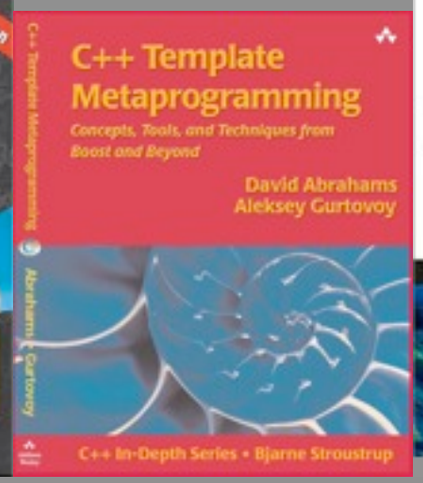

Software Pioneers

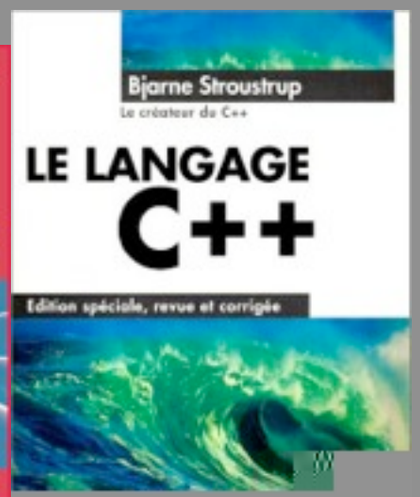

OBJECT-ORIENTED MODELING AND DESIGN


BJARNE STROUSTRUP
The Design and Evolution of
C++

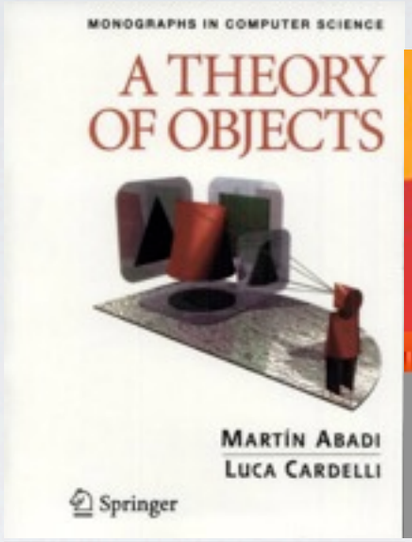
PRACTICAL DEBUGGING IN C++
ANN R. FORD / TOBY J. TEOREY

Claude Delannoy
6e édition
Programmer en langage C++
Borland C++BuilderX Personnel
EYROLLES


C++ Template Metaprogramming
Concepts, Tools, and Techniques from Boost and Beyond
David Abrahams
Aleksey Gurtovoy
C++ In-Depth Series • Bjarne Stroustrup


Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Édition spéciale, revue et corrigée

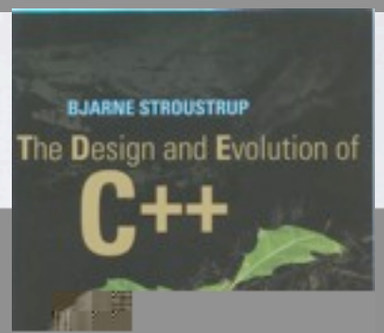
MONOGRAPHS IN COMPUTER SCIENCE

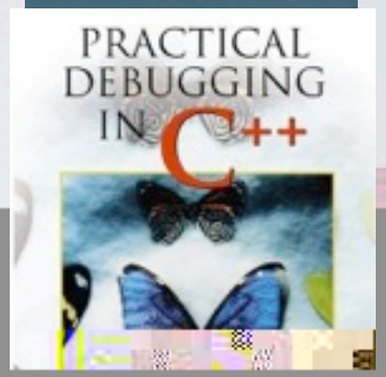

MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer

Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML

Manfred Broy
Ernst Denert (Eds.)
Software Pioneers
Contributions to Software Engineering

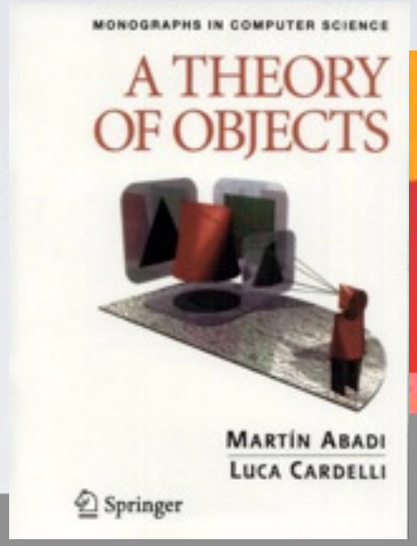
OBJECT-ORIENTED MODELING AND DESIGN


BJARNE STROUSTRUP
The Design and Evolution of C++

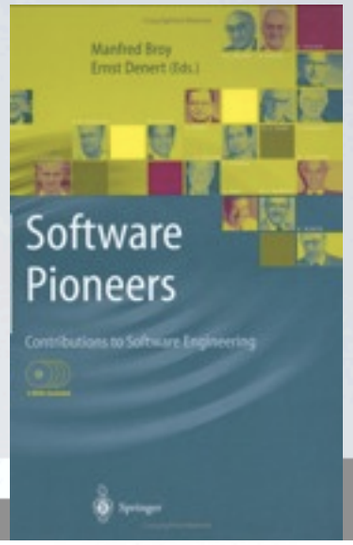

PRACTICAL DEBUGGING IN C++

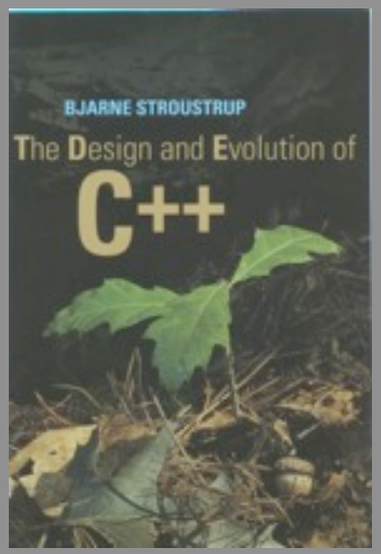
Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Edition spéciale, revue et corrigée

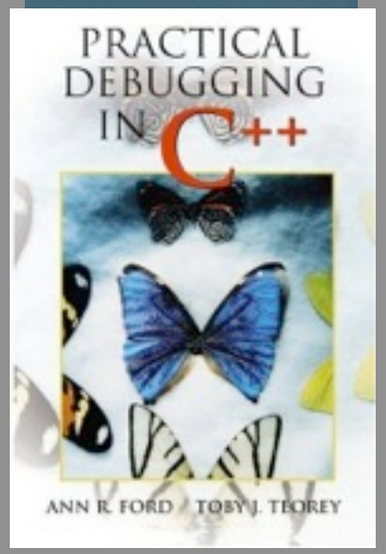
Deuxième
édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet
avec
UML
Sur le CD offert avec ce livre :
Rational Rose 2000 pour Windows 95/98/NT4*
Outils de validation et de prototypage
UML d'ObjeXion Software*
Modèles UML et prototypes des deux études de cas du livre
Spécifications UML 1.3**
Sur le site Web d'accompagnement :
Dialoguez avec les auteurs
Consultez les mises à jour et compléments
www.editions-eyrolles.com
Eyrolles


MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer

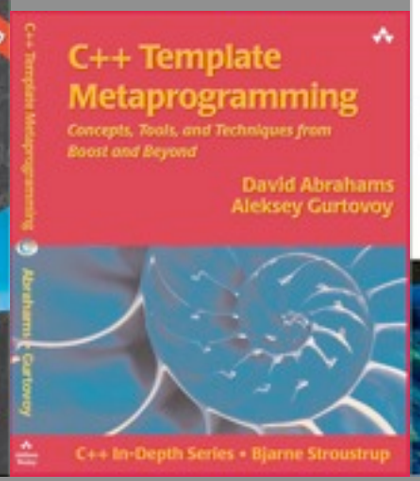
Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML


Software Pioneers

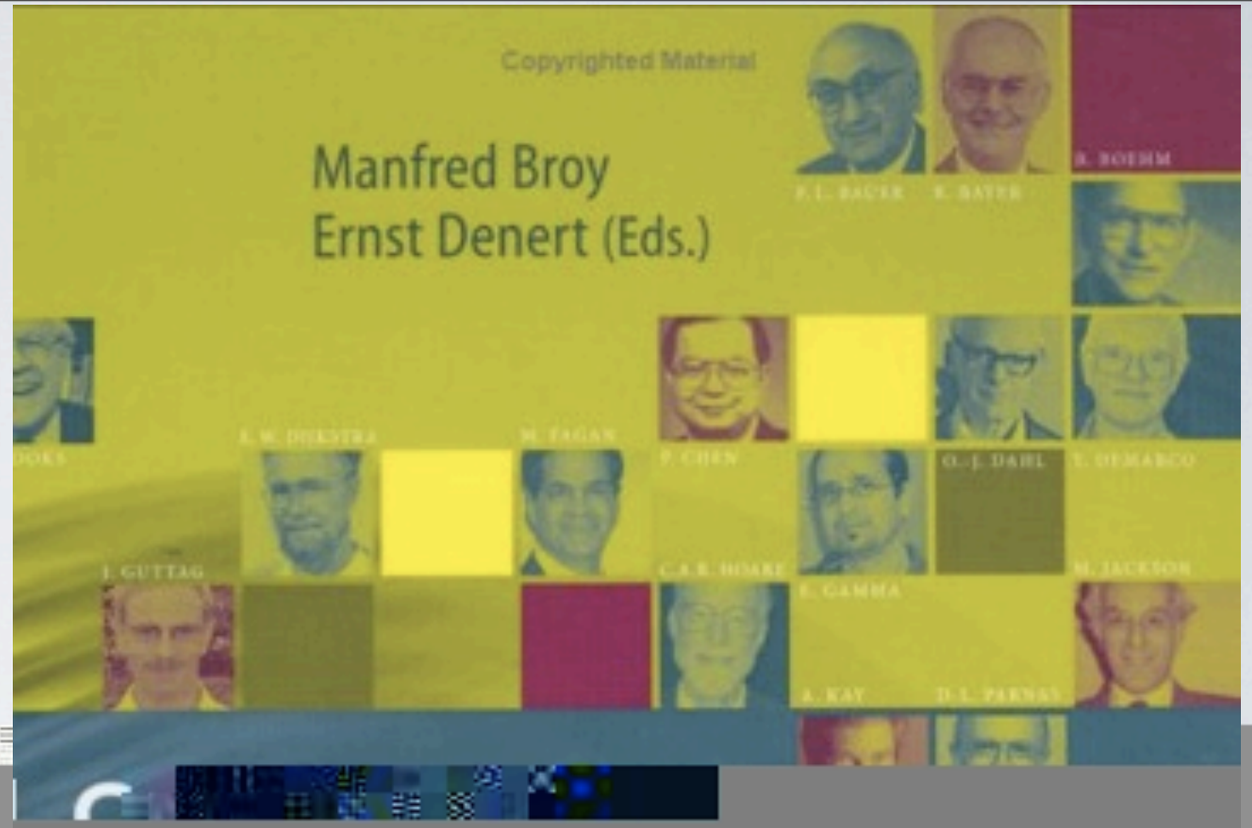

BJARNE STROUSTRUP
The Design and Evolution of
C++


PRACTICAL DEBUGGING IN C++
ANN R. FORD / TOBY J. TEOREY

Claude Delannoy
6e édition
Programmer en langage C++
Borland C++BuilderX Personnel
EYROLLES

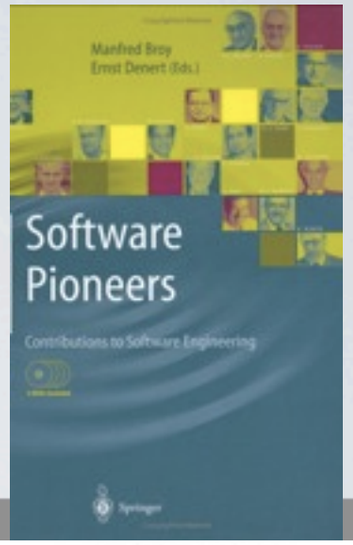

C++ Template Metaprogramming
Concepts, Tools, and Techniques from Boost and Beyond
David Abrahams
Aleksey Gurtovoy
C++ In-Depth Series • Bjarne Stroustrup

Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Édition spéciale, revue et corrigée

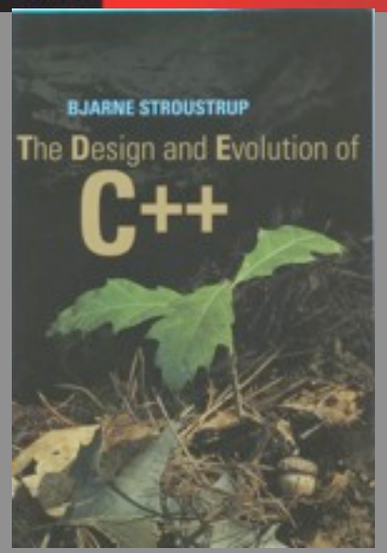

Copyrighted Material
Manfred Broy
Ernst Denert (Eds.)

MONOGRAPHS IN COMPUTER SCIENCE

Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML


Manfred Broy
Ernst Denert (Eds.)
Software Pioneers
Contributions to Software Engineering

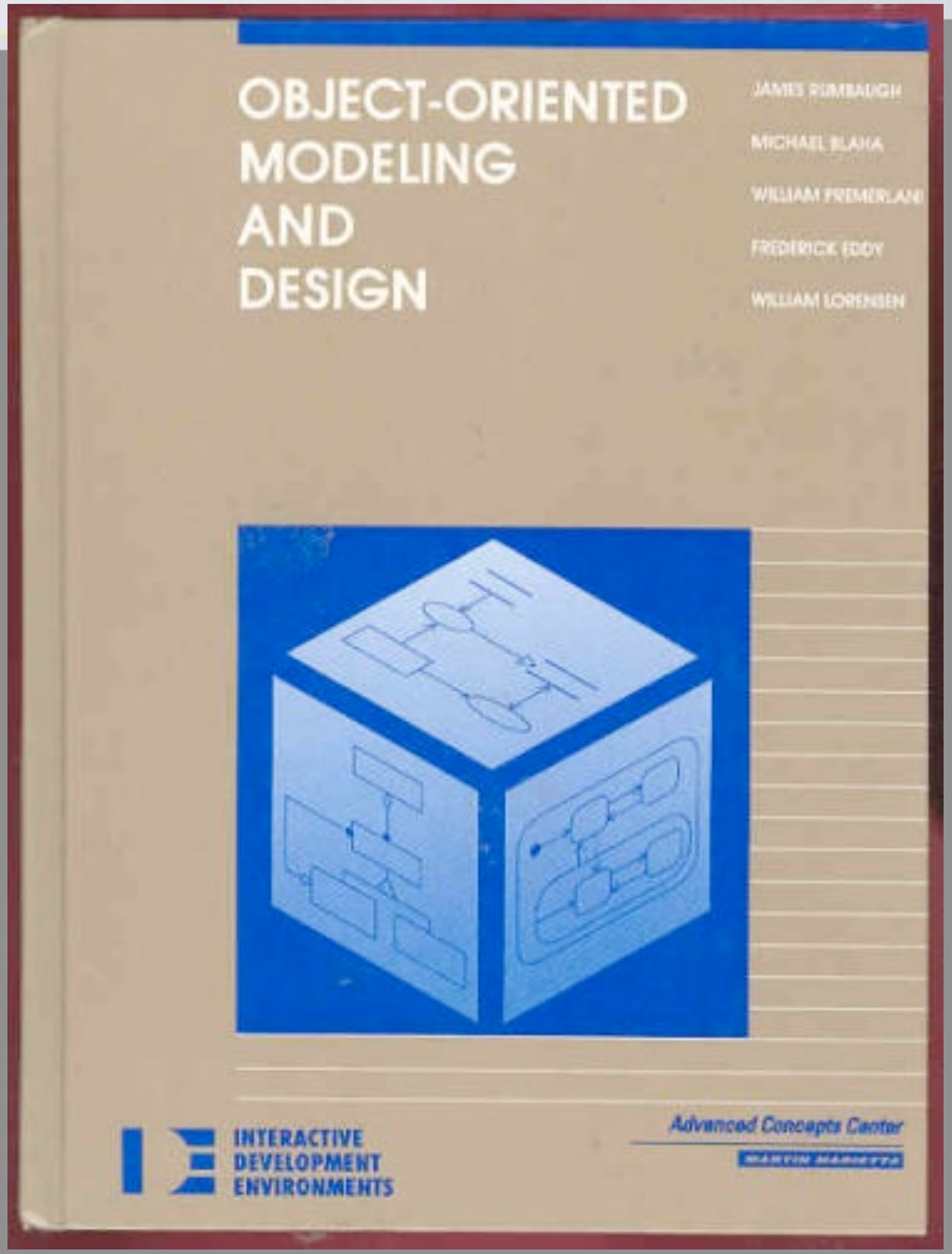
OBJECT-ORIENTED MODELING AND DESIGN

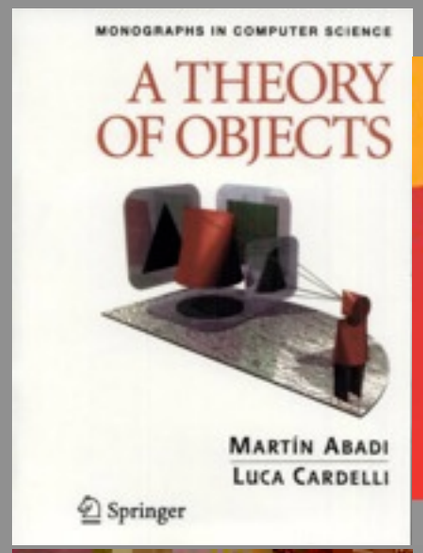

BJARNE STROUSTRUP
The Design and Evolution of
C++

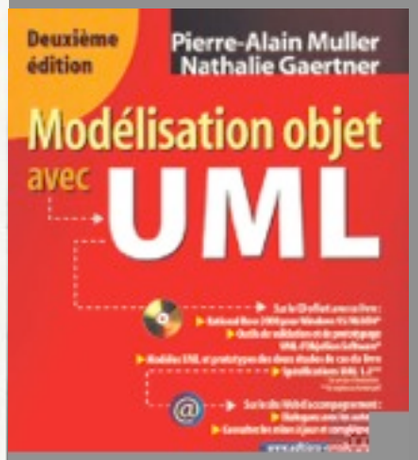
PRACTICAL DEBUGGING IN C++
ANN R. FORD / TOBY J. TEOREY

Claude Delannoy

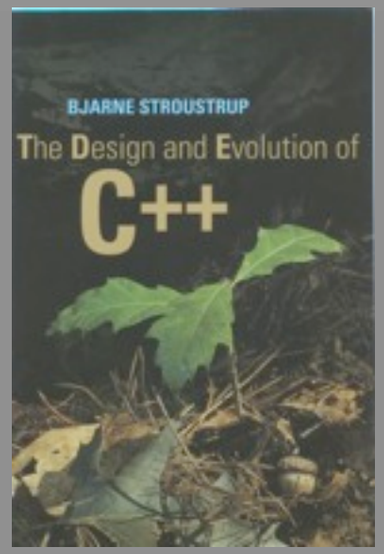
Bjarne Stroustrup
Le créateur du C++
LE LANGAGE
C++
Edition spéciale, revue et corrigée


OBJECT-ORIENTED
MODELING
AND
DESIGN
JAMES RUMBAUGH
MICHAEL BLAHA
WILLIAM PREMERLANI
FREDERICK EDDY
WILLIAM LORENSEN
INTERACTIVE
DEVELOPMENT
ENVIRONMENTS
Advanced Concepts Center


MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer

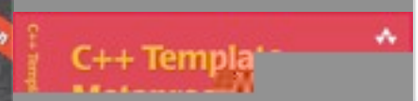

Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML

Software Pioneers
Contributions to Software Engineering

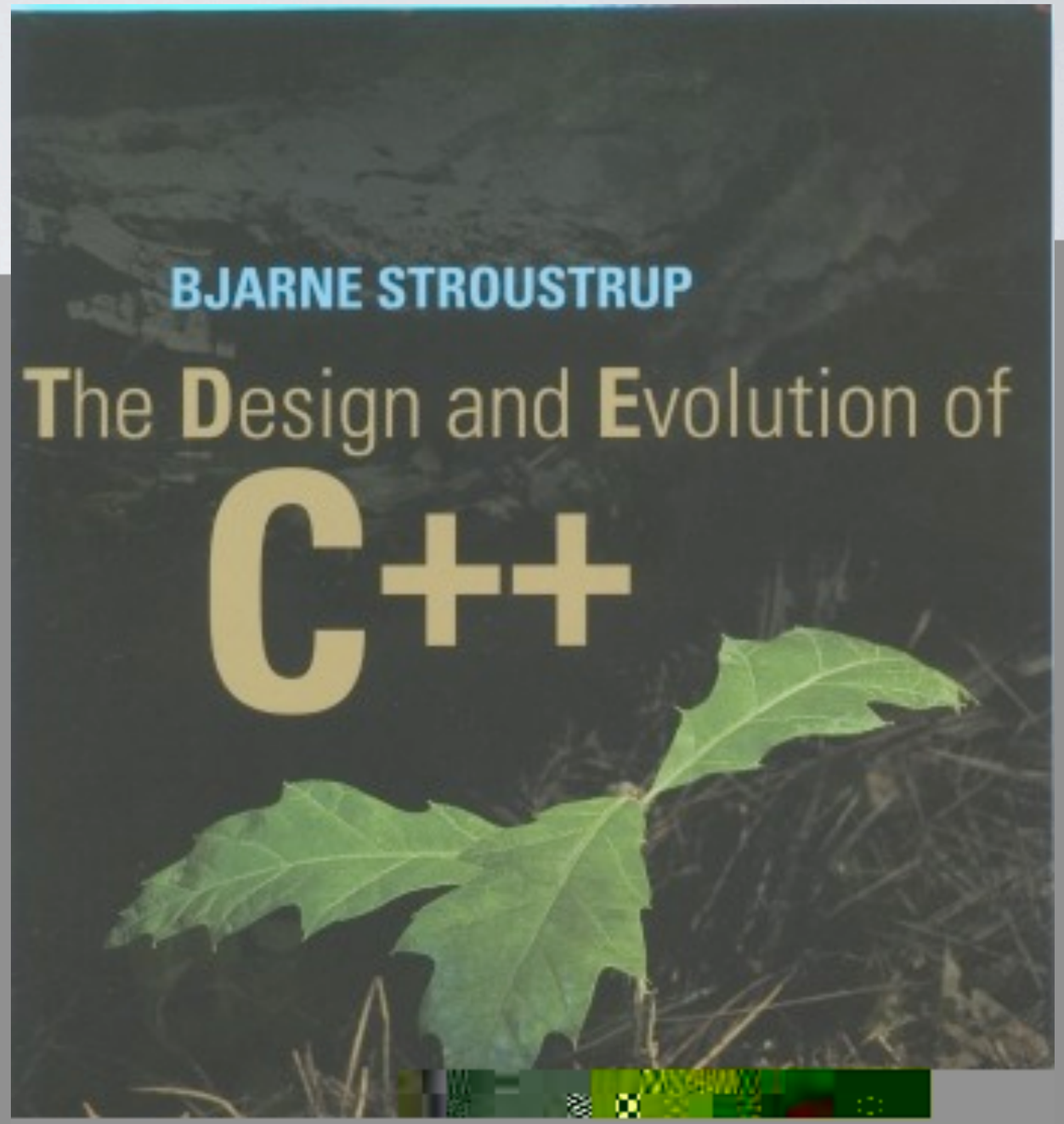

BJARNE STROUSTRUP
The Design and Evolution of
C++

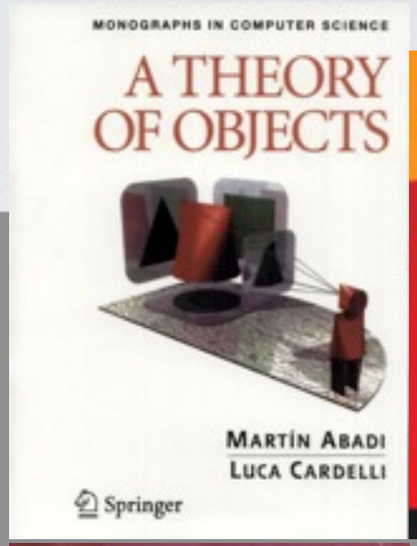
PRACTICAL
DEBUGGING

Claude Delannoy
6e édition
Programmer en langage C++
Borland C++BuilderX Personnel
EYROLLES


C++ Templa

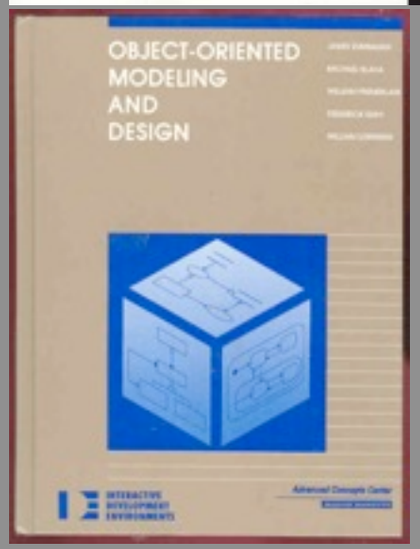
Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Édition spéciale, revue et corrigée

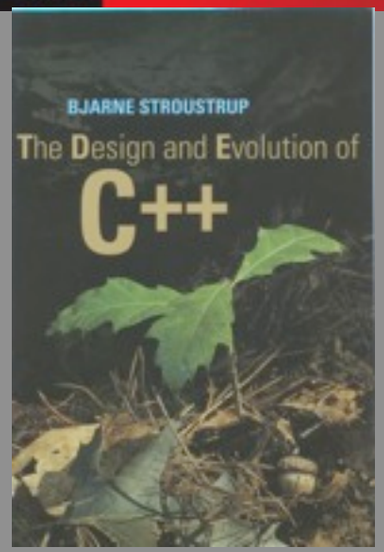

BJARNE STROUSTRUP
The Design and Evolution of
C++


MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer

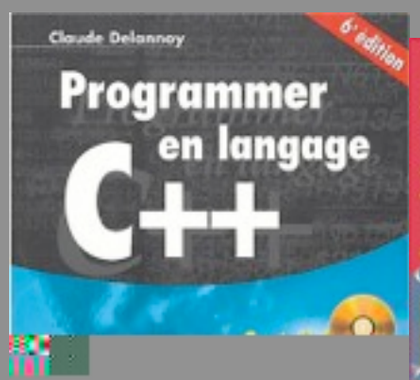
Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML

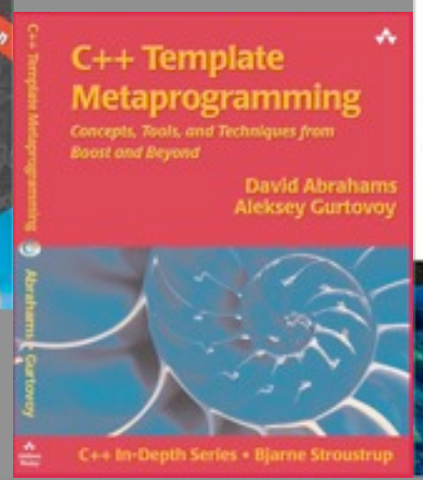
Manfred Broy
Ernst Denert (Eds.)
Software Pioneers
Contributions to Software Engineering


OBJECT-ORIENTED MODELING AND DESIGN


BJARNE STROUSTRUP
The Design and Evolution of C++

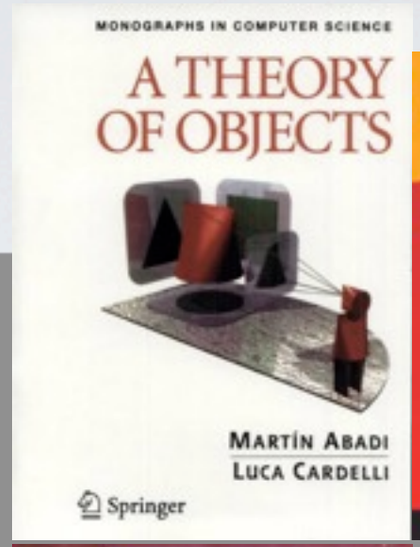
PRACTICAL DEBUGGING IN C++
ANN R. FORD / TOBY J. TEOREY


Claude Delannoy
6e édition
Programmer en langage C++


C++ Template Metaprogramming
Concepts, Tools, and Techniques from Boost and Beyond
David Abrahams
Aleksey Gurtovoy
C++ In-Depth Series • Bjarne Stroustrup

Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Édition spéciale, revue et corrigée

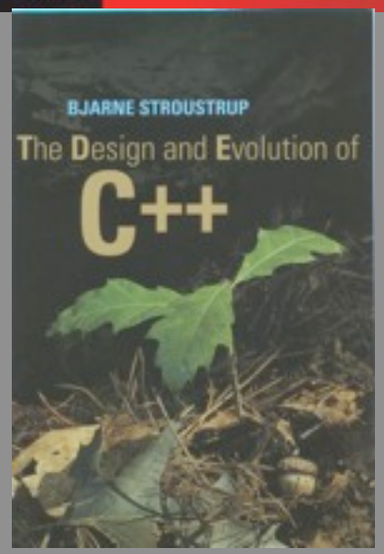
PRACTICAL DEBUGGING

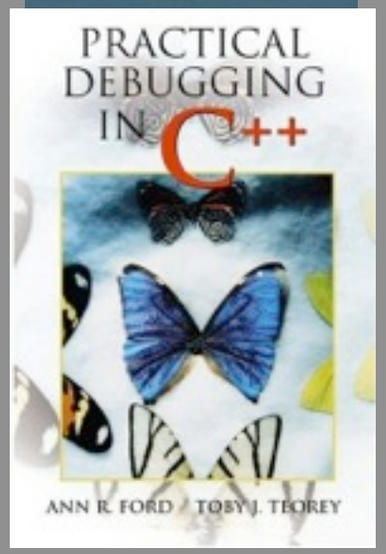

MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer

Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML

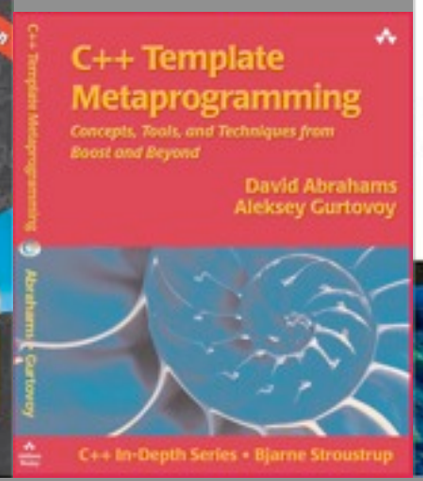
Manfred Broy
Ernst Denert (Eds.)
Software Pioneers
Contributions to Software Engineering

OBJECT-ORIENTED MODELING AND DESIGN

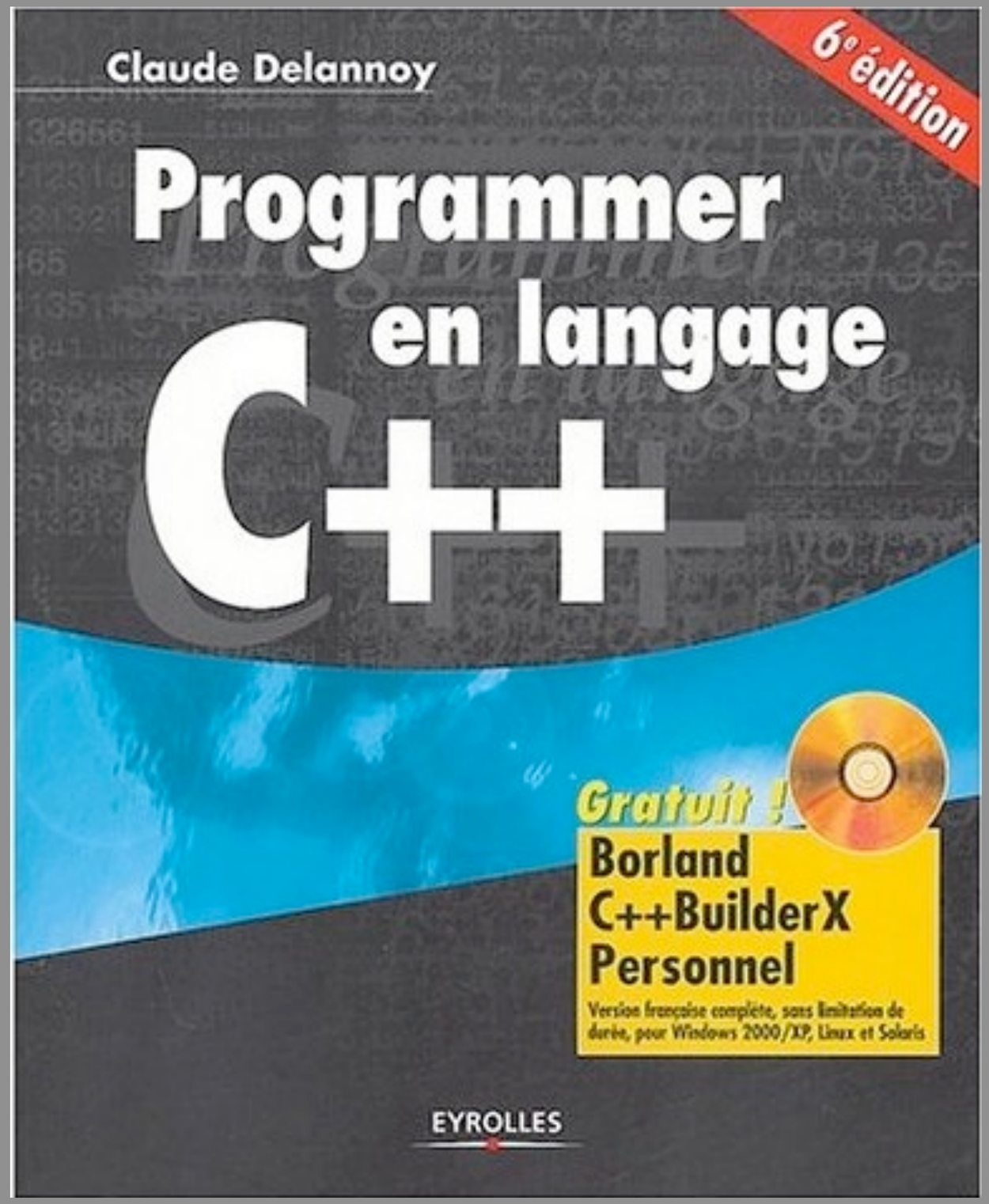

BJARNE STROUSTRUP
The Design and Evolution of C++

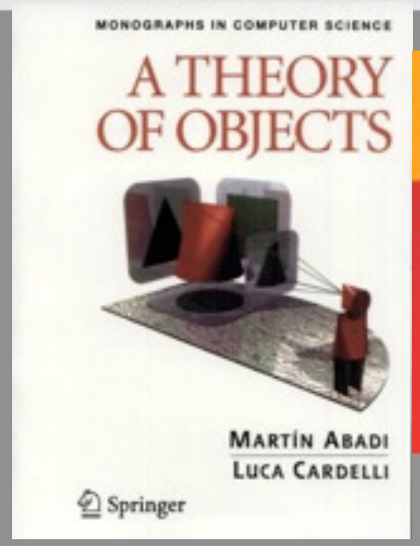

PRACTICAL DEBUGGING IN C++
ANN R. FORD / TOBY J. TEOREY

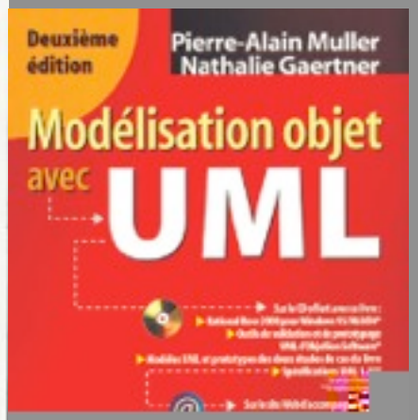
Claude Delannoy
6e édition
Programmer en langage C++
Borland C++BuilderX Personnel
EYROLLES


C++ Template Metaprogramming
Concepts, Tools, and Techniques from Boost and Beyond
David Abrahams
Aleksey Gurtovoy
C++ In-Depth Series • Bjarne Stroustrup

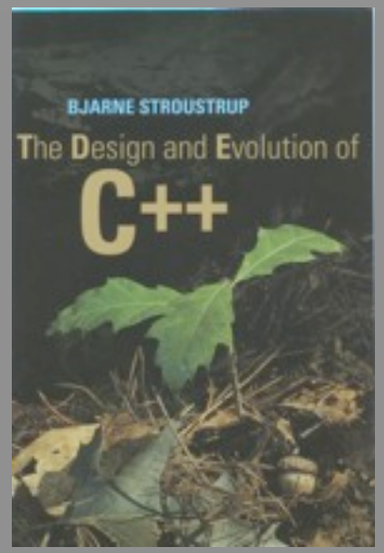
Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Édition spéciale, revue et corrigée


Claude Delannoy
6e édition
Programmer en langage C++
Gratuit !
Borland C++BuilderX Personnel
Version française complète, sans limitation de durée, pour Windows 2000/XP, Linux et Solaris
EYROLLES

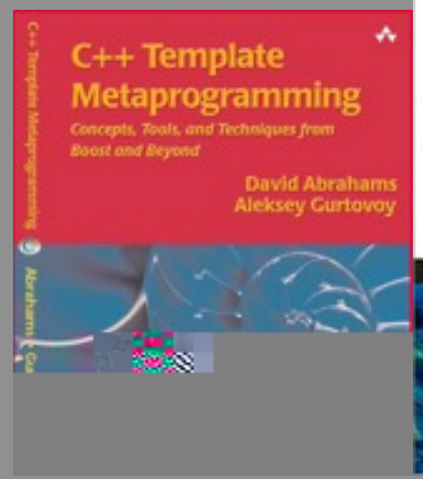

MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer


Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML

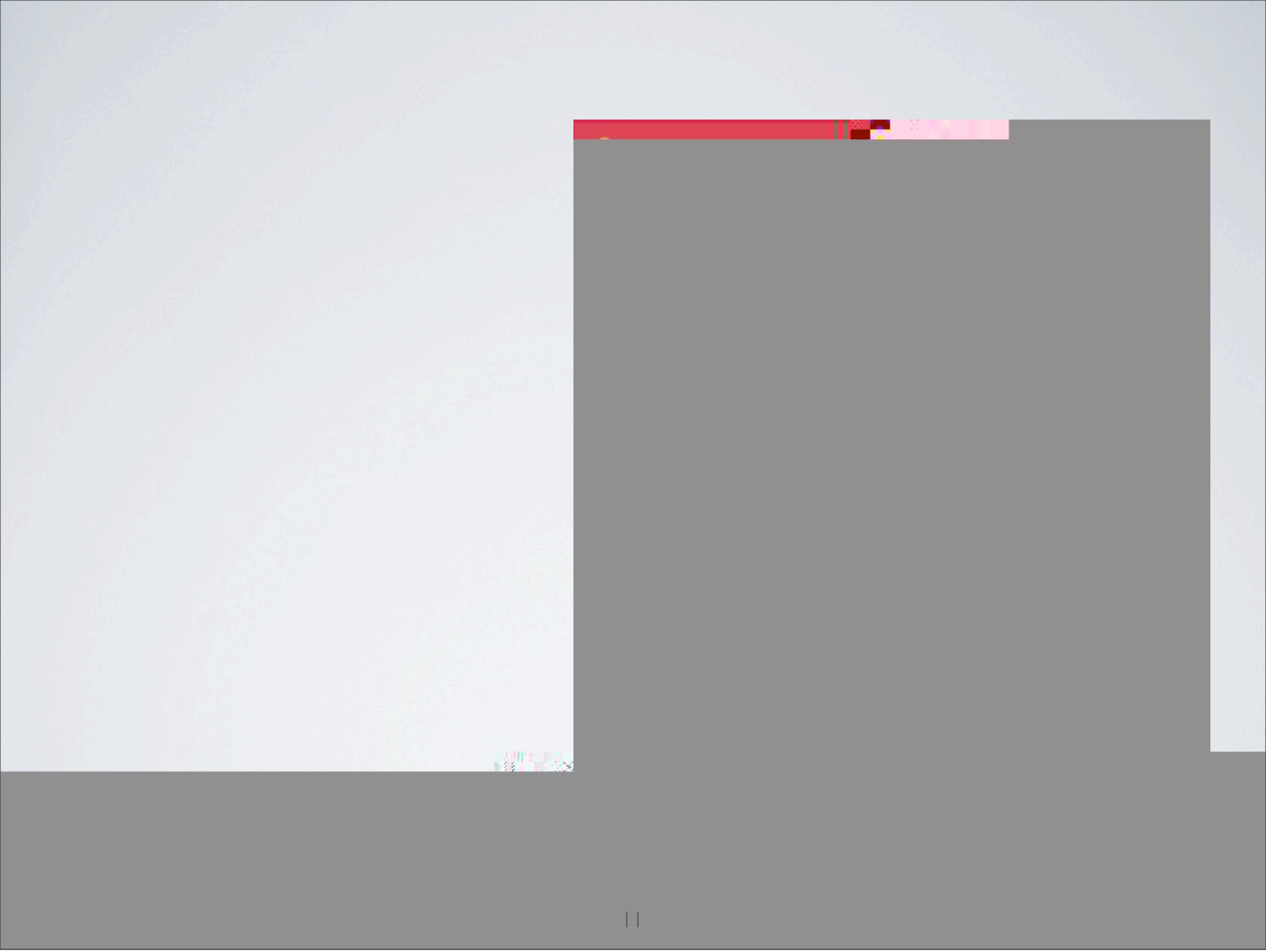
Software Pioneers

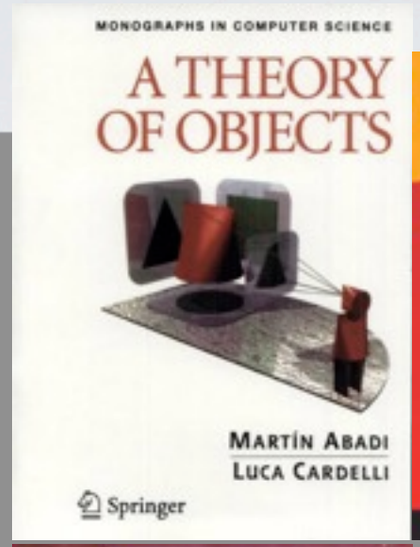

BJARNE STROUSTRUP
The Design and Evolution of C++

PRACTICAL DEBUGGING IN C++


C++ Template Metaprogramming
Concepts, Tools, and Techniques from Boost and Beyond
David Abrahams
Aleksey Gurtovoy

Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Édition spéciale, revue et corrigée

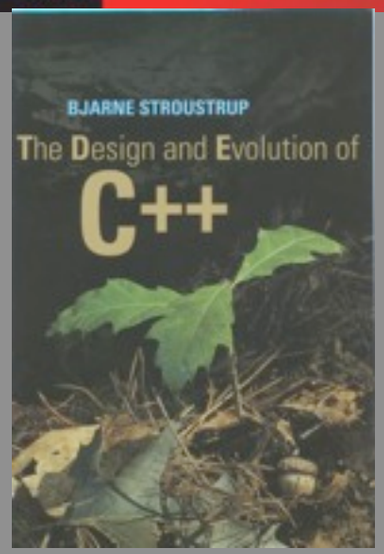


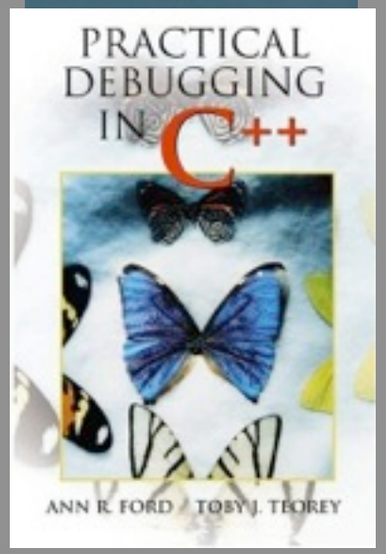

MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer

Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML

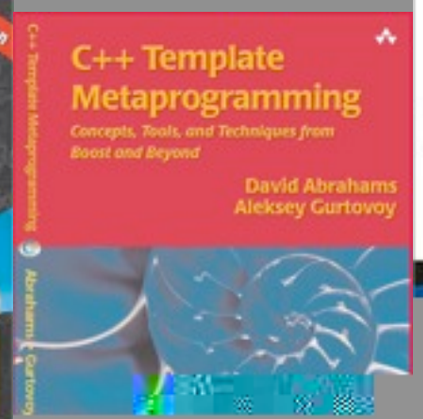
Manfred Broy
Ernst Denert (Eds.)
Software Pioneers
Contributions to Software Engineering

OBJECT-ORIENTED MODELING AND DESIGN

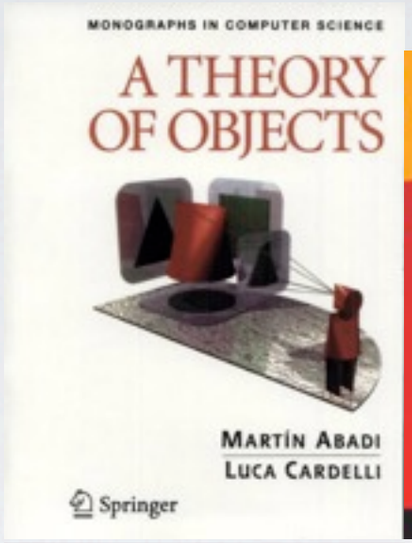

BJARNE STROUSTRUP
The Design and Evolution of C++


PRACTICAL DEBUGGING IN C++
ANN R. FORD / TOBY J. TEOREY

Claude Delannoy
6e édition
Programmer en langage C++
Borland C++BuilderX Personnel
EYROLLES


C++ Template Metaprogramming
Concepts, Tools, and Techniques from Boost and Beyond
David Abrahams
Aleksey Gurtovoy

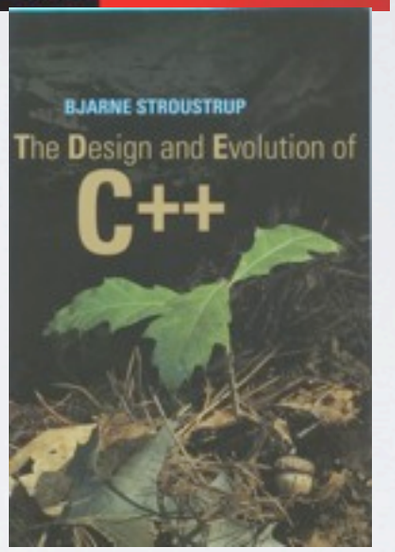
Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Édition spéciale

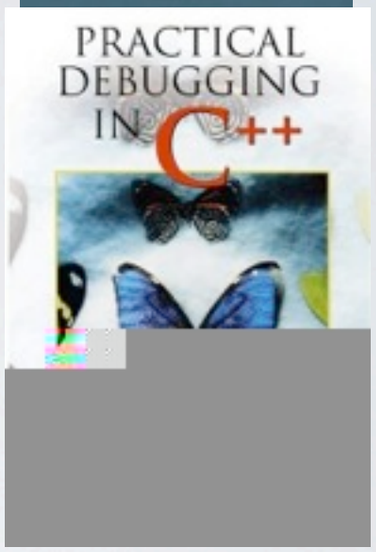

MONOGRAPHS IN COMPUTER SCIENCE
A THEORY OF OBJECTS
MARTÍN ABADI
LUCA CARDELLI
Springer

Deuxième édition
Pierre-Alain Muller
Nathalie Gaertner
Modélisation objet avec UML

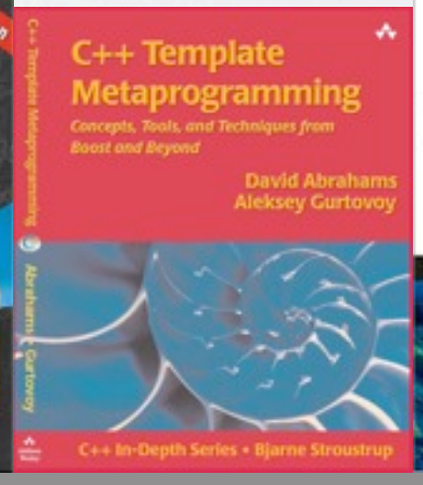
Manfred Broy
Ernst Denert (Eds.)
Software Pioneers
Contributions to Software Engineering

OBJECT-ORIENTED MODELING AND DESIGN


BJARNE STROUSTRUP
The Design and Evolution of C++


PRACTICAL DEBUGGING IN C++

Claude Delannoy
6e édition
Programmer en langage C++
Gratuit !
Borland C++BuilderX Personnel
EYROLLES


C++ Template Metaprogramming
Concepts, Tools, and Techniques from Boost and Beyond
David Abrahams
Aleksey Gurtovoy
C++ In-Depth Series • Bjarne Stroustrup

Bjarne Stroustrup
Le créateur du C++
LE LANGAGE C++
Édition spéciale, revue et corrigée

INTRODUCTION

Deux visions (compatibles) de la question

- Une extension du langage C ?
- Un langage orienté objet à la syntaxe inspirée du C ?

Deux approches pédagogiques

- *bottom-up*
- *top-down*

Le C++ repose sur **un modèle d'objet**

Encapsulation

- droits d'accès

Instanciation de classe

- construction/destruction
- statique vs dynamique

Messages

- invocation de méthodes

Généralisation

- héritage, redéfinition, extension
- types abstraits, factorisation, méthodes virtuelles

On y distingue **trois polymorphismes**

- *ad-hoc* (la surcharge)
- par sous-typage (l'héritage)
- généralisé ou universel (les modèles)

La généricité :

- fonctions/méthodes génériques
- types génériques (aka templates)

Rappels historiques à propos du langage C++ ou une ode à la gloire de *(Bell)* :

- 1979-1982 : C avec classes
- 1983-1985 : C++, cfront, manuel de référence
- 1986-1988 : diffusion importante du langage
- 1989-1998 : cfront 2.0, cfront 3.0, STL (Alexander Stepanov - HP), ISO 14882:1998
- 2003 : ISO 14882:2003
- 2011 : ISO 14882:2011 *aka* C++11

- `isocpp.org` (`/std`)

- Attention le standard est un document de 1400 pages!!!

Langages ayant influencé la conception du C++

C (1972, Kernighan & Ritchie - Bell)

Langages ayant influencé la conception du C++

Simula (1960, Dahl & Nygaard - Oslo)

Langages ayant influencé la conception du C++

Ada 83 (1980, Ichbiah - CII-Honeywell Bull)

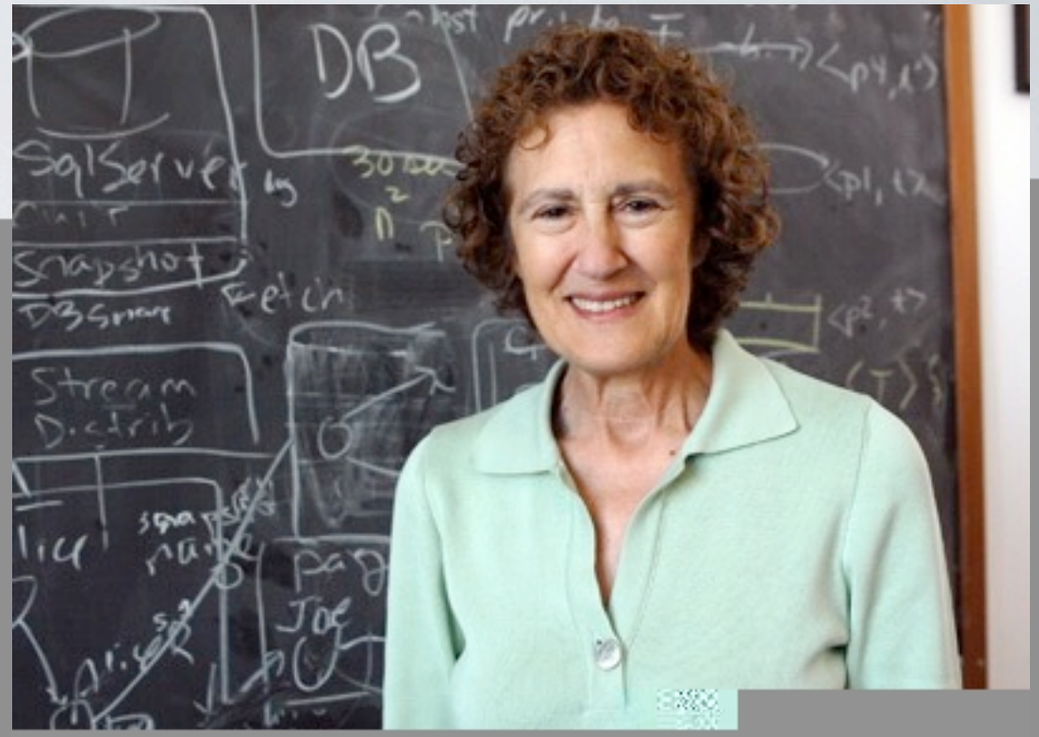
†

Langages ayant influencé la conception du C++

Algol 68 (inspiré d'Algol 60, Backus & Naur)

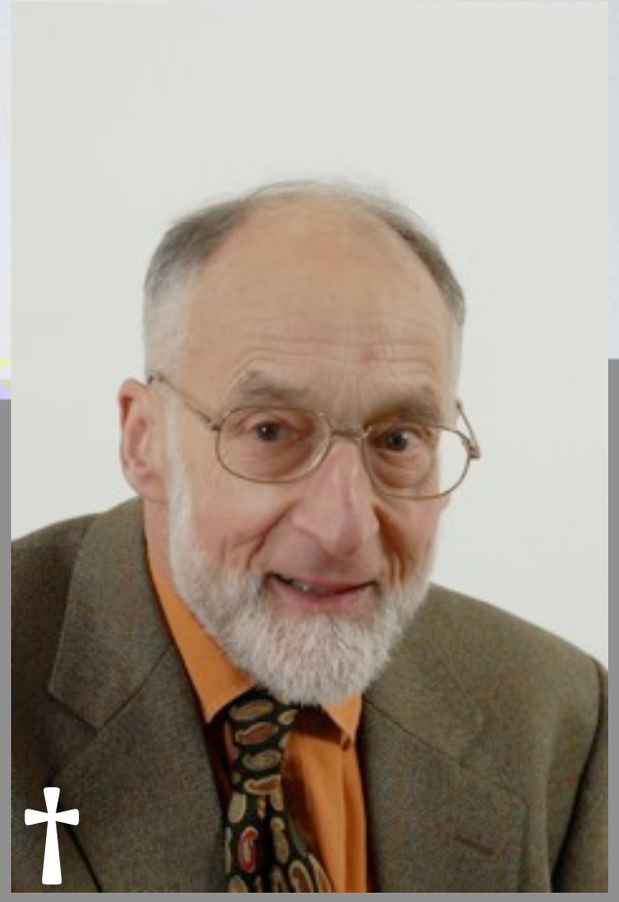
Langages ayant influencé la conception du C++

CLU (1975, Liskov - MIT)



Langages ayant influencé la conception du C++

ML (1980, Milner - Edimbourg)



Quelques caractéristiques :

- compatibilité totale avec le C (sauf de très rares exceptions bien connues, faciles à identifier et à réparer)
- performances comparables à celles du C (une obsession de conception) : on *colle* à la machine

Des **critiques**. « C++ **sucks** » :

At a recent conference, a speaker asked for a show of hands and found that twice as many people claimed to hate C++ as had ever written even a single small C++ program. The only word for such behavior is bigotry. In dealing with the current wave of C++ bashing, we should remember that bigotry is bred by ignorance and fear.

Des **critiques**. « C++ is too **complex** » :

Experience shows that trying to learn OOP or C++ from The Annotated C++ Reference Manual is usually a BIG mistake. It is a nice book, I can recommend it (:-), but not as a tutorial. After all, most people wouldn't try learning a natural language exclusively from a grammar plus a dictionary. Few would succeed.

Des **critiques**. « C++ programmers are **idiots** »

Many of the most experienced individuals and organizations in the industry (and academia) use C++. Ascribing their choice of C++ over alternatives to stupidity, ignorance, or inexperience is arrogant and insulting - and in many cases simply wishful thinking based on a limited field of knowledge. I still think the average C++ programmer is pretty smart, but rarely a fanatic, and usually too busy getting the work done to bother with language laws or the latest academic advances in OO type theory. Fortunately, one doesn't need to be a genius to write good C++.

Des **critiques**. « C++ is **useless**/**unreliable**/ **dangerous** because it lacks feature/property X » :

My firm conclusion is that no single feature is truly necessary. Much more successful software has been written in languages proclaimed BAD, than has been written in languages acclaimed as saviors of suffering programmers; much more.

Des **critiques**. « C++ is **not Object-Oriented** »

C++ supports object-oriented programming (as defined by most people). C++ supports it pretty well for real-world applications. Its type system and general model of the world are coherent and relevant to real applications.

The proof of the pudding is in the eating, so I consider the fact that many large projects have been completed using C++ much more significant than the fact that C++ doesn't conform to the latest fashion of OO theology.

Have a look at the "Design Patterns" book by Gamma et. al. for examples of elegant designs clearly expressed in relatively straightforward C++. I observe that there are a lot of myths about what features C++ has and doesn't have.

Un exemple original :

Un exemple :

Un exemple :

Un exemple :

Un exemple :

Un exemple :

Un exemple :