

Conception et programmation orientées objet

Bertrand Meyer

Traduit de l'anglais par Pierre Jouvelot

© Groupe Eyrolles, 2000, pour le texte de la présente édition en langue française.

© Groupe Eyrolles, 2008, pour la nouvelle présentation, ISBN : 978-2-212-12270-1

EYROLLES



Préface de la seconde édition

De nombreux événements se sont produits dans le monde orienté objet depuis la première édition d'*OOSC* (le nom sous lequel ce livre (NdT : sa version anglaise) est désormais connu) publiée en 1988. L'explosion de l'intérêt pour la technologie OO suggéré dans la préface de la première édition, reproduite dans les pages qui précèdent sous une forme légèrement étendue, n'était rien en comparaison de ce à quoi nous avons assisté depuis. De nombreux journaux et conférences sont aujourd'hui dédiés à la technologie objet ; Prentice Hall (NdT : éditeur de la version anglaise) propose une série entière de livres consacrés au sujet ; des développements marquants se sont produits dans des domaines comme les interfaces utilisateur, la concurrence et les bases de données ; de nouveaux sujets d'étude ont émergé, comme l'analyse OO et les spécifications formelles ; le calcul réparti, qui était autrefois un sujet spécialisé, devient de plus en plus d'actualité dans les développements actuels, en partie grâce à la croissance d'Internet ; et le Web influence le travail quotidien de tout un chacun.

Et les bonnes nouvelles ne s'arrêtent pas là. Le progrès constant du monde du logiciel fait plaisir à voir — en partie grâce à la diffusion encore incomplète mais irréfutable de la technologie objet. Trop nombreux sont les livres et articles consacrés au génie logiciel qui débudent encore par les lamentations obligées sur la “crise du logiciel” et l'état pitoyable de notre industrie comparée aux *vraies* disciplines d'ingénierie (qui, comme nous le savons tous, ne ratent jamais rien). Un tel aveu n'est pas de mise. Oh, nous avons encore bien du chemin à parcourir, comme le savent trop bien tous ceux qui utilisent des produits logiciels. Mais, étant donné les défis auxquels nous devons faire face, il n'y a pas de raison d'avoir honte de nous ; et nous nous améliorons à chaque instant. L'ambition de ce livre, comme celle de son prédécesseur, est de faciliter ce processus.

Cette seconde édition n'est pas une mise à jour, mais le résultat d'une profonde refonte. Il n'est pas un paragraphe de la version d'origine qui n'ait pas été modifié. (En fait, presque pas une seule ligne.) De nouveaux sujets ont été rajoutés, y compris un chapitre complet sur la concurrence, la répartition, le calcul client-serveur et la programmation Internet ; un autre sur la persistance et les bases de données ; un sur les interfaces utilisateur ; un sur le cycle de vie du logiciel ; de nombreux schémas de conception et techniques d'implémentation ; une exploration en profondeur d'un aspect méthodologique peu documenté auparavant : comment utiliser au mieux, ou ne pas utiliser, l'héritage et éviter de l'utiliser de travers ; des discussions sur de nombreux autres sujets de méthodologie orientée objet ; une présentation extensive de la théorie des types abstraits de données — le fondement mathématique de notre sujet, indispensable à toute compréhension en profondeur de la technologie objet, et rarement présenté en détail dans les livres d'enseignement et les tutoriels ; une présentation de l'analyse OO ; des centaines de nouvelles références bibliographiques et de sites Web ; la description d'un environnement orienté objet complet et de ses concepts sous-jacents ; et des tonnes de nouveaux principes, idées, avertissements, explications, figures, exemples, comparaisons, citations, classes et routines.

Les réactions à *OOSC-1* ont été si gratifiantes que je sais que les lecteurs attendent beaucoup. J'espère que ce qu'ils trouveront dans *OOSC-2* les intriguera, leur sera utile et répondra à leurs attentes.

Santa Barbara B.M.
Janvier 1997