

Book Review

The Unified Modeling Language Reference Manual, Second Edition

by James Rumbaugh, Ivar Jacobson and Grady Booch, Addison-Wesley,
Boston, Massachusetts, 2005. 721 pp., \$64.00. ISBN 0321245628.

Reviewed by Charles Ashbacher

Like all dynamic languages, the Unified Modeling Language (UML) is growing more complex over time. While it is true that for most developers, this means that you will regularly use a smaller percentage of the language, the actual percentage will vary from person to person and from day to day. Therefore, no abridged UML manual could possibly be adequate. Written by the three creators of the UML, this manual is clearly definitive and one that all developers should have at extended arms reach. Designed to cover the changes in the recently released UML 2.0, which were significant, a CD with the full text in Adobe PDF form with hotlinks to the definitions of the key terms is also included.

The opening chapter is an overview of the UML and most people can skip it. Chapter two is an overview of models, and this one is worth reading. Short, it introduces some of the fundamental terminology and approaches. A walkthrough of UML is done in chapter three, which introduces the various formal views of a project. They are: static, design, use case, state machine, activity, interaction, deployment, and model management. Each of these views is then explained in a short chapter. These chapters should be required reading for users of the book, as they establish much of the notational and definitional background used in the reference section.

The real value of the book is in the five hundred plus pages of detailed definitions of the key terms and phrases in the UML. Listed in alphabetical order, each entry has the following form:

- *) Entry name: the term or phrase.
- *) A brief definition, generally one or two sentences.

- *) The semantics of the term, generally using several paragraphs. This section includes the structure, subordinate items and often an example.
- *) The notation of usage. Options and guidelines for use are often included.
- *) Discussion (occasional), where the author's opinions and/or a background explanation of the term are given.
- *) History (where appropriate), the changes in how the term is interpreted from earlier versions of the UML.

Quite frankly, I cannot see how it would be possible for any developer to use anything more than a very tiny subset of the UML if they do not have access to this book. All speakers of a language can use that language in informal communication, but when we want to communicate ideas formally and precisely, a dictionary is essential. That is the role that this book will fill, as no human communication is more precise than when we do it with notations that describe software.