The Fraunhofer IESE Series on Software and Systems Engineering

Jürgen Münch
Ove Armbrust
Martin Kowalczyk
Martín Soto

Software Process Definition and Management

Fraunhofer IESE — Springer

2012, XX, 236 p.

📙 Printed book

Hardcover
► 69,95 € | £62.99 | $89.95
► *74,85 € (D) | 76,95 € (A) | CHF 93.50

🄴 eBook

Available from your library or
► **springer.com/shop**

Ⓜ MyCopy

Printed eBook for just
► € | $ 24.99
► **springer.com/mycopy**

J. **Münch**, O. **Armbrust**, M. **Kowalczyk**, M. **Soto**
Software Process Definition and Management
Series: The Fraunhofer IESE Series on Software and Systems Engineering

► **First book that gives a comprehensive introduction into software process definition and management**
► **It enables readers to develop useful process models that are suitable for their own purposes**
► **Special chapters on measurement and determination of the effectiveness of processes**

The concept of processes is at the heart of software and systems engineering. Software process models integrate software engineering methods and techniques and are the basis for managing large-scale software and IT projects. High product quality routinely results from high process quality.

Software process management deals with getting and maintaining control over processes and their evolution. Becoming acquainted with existing software process models is not enough, though. It is important to understand how to select, define, manage, deploy, evaluate, and systematically evolve software process models so that they suitably address the problems, applications, and environments to which they are applied. Providing basic knowledge for these important tasks is the main goal of this textbook.

Münch and his co-authors aim at providing knowledge that enables readers to develop useful process models that are suitable for their own purposes. They start with the basic concepts. Subsequently, existing representative process models are introduced, followed by a description of how to create individual models and the necessary means for doing so (i.e., notations and tools). Lastly, different possible usage scenarios for process management are highlighted (e.g. process improvement and software process simulation).

Their book is aimed at students and researchers working on software project management, software quality assurance, and software measurement; and at practitioners who are interested in process definition and management for developing, maintaining, and operating software-intensive systems and services.