To deliver on its promises, object-oriented technology must be able to produce systems that are reliable (correct and robust). Only under these conditions will the other quality factors advertised for the method, in particular the increase in reusability and extendibility, yield the expected benefits for software practitioners. It is indeed possible to use object-oriented technology to produce, almost routinely, software systems that reach a degree of reliability without any equivalent in conventional methods, languages and tools. This requires a strict application of pure object-oriented concepts, in particular, seamlessness (the use of a consistent method and notation throughout the software life-cycle), information hiding, automatic garbage collection, static typing, and the combination of static typing and dynamic binding. Another key component of this approach is the notion of design by contract, which leads to the production of systems whose correctness is built-in rather than ensured ex post facto.

Through a number of examples, the presentation will explain the theory of Design by Contract and its applications. It will show how the ideas lead to sound architectural principles in the design of large-scale systems, make it possible to control the use of inheritance, and yield a disciplined approach to the handling of abnormal cases.

Bertrand Meyer is president of Interactive Software Engineering Inc. and a pioneer of object technology through his books, in particular, *Object-Oriented Software Construction* (whose second edition has just been published by Prentice Hall), *Reusable Software* and *Object Success*. Active in both the business and the academic scenes, he has directed the development of widely used O-O tools and libraries totaling hundreds of thousands of lines and taught O-O principles and modern software engineering worldwide. He is editor of the Object Technology column of *IEEE Computer*, the Prentice Hall O-O Series, and the Addison-Wesley *Eiffel in Practice Series*. 