Achieving the correctness of software as an ascertained fact is an ambitious goal for software development methods that is not reached so far by the methods applied in the practice. This goal can only be achieved by a method that is consequently directed towards securing correctness in all phases.

Such a method has been developed by the KORSO project. It is strictly based on the concept of axiomatic specification and covers all phases of software development from requirement capture to requirement specification, design and implementation.

The first paper, A Method for the Development of Correct Software, follows on directly from the introductory papers, describing a method based on axiomatic specifications. The model of a development graph and its handling and manipulation are described in technical terms. This method is applied in the next paper Realizing Sets by Hash Tables which looks at a special refinement problem, and demonstrates the solution with the KORSO development graph.

The third paper, Event Automata as a Generic Model of Reactive
Systems, gives a mathematical framework with various instantiations for concurrency semantics or object-oriented methods. This is a solid mathematical base for object-oriented methods, as described in On Object-Oriented Design and Verification, which also focuses on the important point of verification. An other view of method support is given in the paper Design of Modular Software Systems with Reuse which looks specifically at the concept of reuse, describing a process model in which existing and implemented modules can be integrated as early as in the specification phase.

The last paper of this part, AVL Trees Revisited: A Case Study in SPECTRUM shows a pass through the KORSO method, using the specification language SPECTRUM. It demonstrates deductive software development starting from an axiomatic requirement specification and leading to an executable specification. All proof obligations involved in the development process are indicated.