In high-performance computing, the available computing resources shall be utilized as fully as possible. This typically requires restructuring of algorithms throughout a program’s architecture. The goal of this thesis is to develop strategies for quick experiments with different architectural building blocks.

Examples of such performance-oriented modifications are:

- Move if statements out of loops if possible.
- Exchange library and function calls with alternative implementations. This includes adaptations of parameters and typically affects related code.
- Improve memory access patterns by rearranging declarations, including adaptations of the code that accesses the memory structures.

The goal of this thesis is to develop a tool that investigates areas where such changes can be applied. The tool shall systematically apply combinations of possible changes in order to find highly efficient configurations. In order to collect information about the effects on performance, the tool shall automate the process of performing run-time experiments; it is expected to deliver insights on the effects of certain modifications.