For every piece of software there is always room for improvement. The driver for these improvements may be for example better understandability and extensibility of the software code. However, it is important that these improvements can be done in a reliable manner so that the existing software behaviour is preserved and the risk of introducing new faults is minimized. The continuous software improvement should be a part of the software process; otherwise the software quality suffers as new functionality is constructed on top of the existing software code, eventually making the software difficult or even impossible to manage.

This Thesis explores the formally defined software restructuring methods and the benefits gained from using them. The study of these restructurings is initiated with a literary review about the existing formalized restructuring methods: the most interesting existing method, refactoring, is described in greater details, mainly based on the Martin Fowler’s *Refactoring – Improving the Design of Existing Code* book.

Although the concept of refactoring is found to be useful in many ways in improving the software quality, some principles of the refactoring are deemed to be too restrictive, especially when specific design frameworks are used: a definition for a new concept of *formalized restructuring* is suggested as a more general formal software restructuring method. A set of suggested formalized restructurings for Ericsson software development environment is presented in this Thesis as an example of the defined concept.

As a conclusion to this Thesis refactoring, or more generally, formalized restructuring methods are recommended to be taken into use inside the design processes; besides the adaptation within the processes, this change requires adaptations in the way of thinking as well. It is also recommended that the habit of formally cataloguing the commonly used restructurings inside different software development environments to ease the actual restructuring processes is taken into use.

Along with the identified benefits of the formal restructuring methods it was discovered during this Thesis that this problem area has still many interesting uncharted areas that require further studying: for example the topics of metrics based refactoring and quality metrics based on bad smells, to mention a few.

Key words: software restructuring, formal restructuring, formalized restructuring, refactoring, Ericsson