# Einführung Version 1.0

## Problemstellung

Die Probleme moderner Software sind sehr vielseitig. Der Kunde hat einen Fehler gefunden und dieser sollte innerhalb kürzester Zeit behoben sein. Ein weiterer Kunde möchte eine Einstellung zum ausblenden zahlreicher Details in einem Report haben. Der dritte Kunde beschwert sich, dass der Dialog welchen er mehrmals am Tag öffnet viel zu lange lädt und er dadurch sehr viel Zeit verliert. Bei all diesen Problemen gibt es einen gemeinsamen Nenner. Die Lösung dieser Probleme erfordert eine Änderung der Codebasis. Bei Änderungen der aktuellen Codebasis gibt es eine zentrale Frage.

* Wo genau muss die Änderung stattfinden?

Diese lässt sich meist nicht so leicht beantworten. Es wird nun ein Beispiel erläutert, welches sich so in jeder beliebigen Softwarefirmen zutragen könnte:

In diesem Szenario wurde beim Design des Programmes darauf geachtet die einzelnen Teile in Module und Klassen zu trennen, jedoch sind die Namen der Methoden sehr unglücklich gewählt und es kann daher ohne eine Debugging Sitzung nicht festgestellt werden welche Methoden aufgerufen werden. Es gibt aber einen kleinen Hinweis bei einer der Methoden in der Klasse. Ein Kommentar der uns sagt, dass in dieser Methode sehr viel Zeit für das Umrechnen der Einheiten benötigt wird. Der Inhalt der Methode lässt darauf schließen, dass der Kommentar schlichtweg falsch ist. Ein Blick in die „History“ unserer Quellcodeverwaltung verrät uns, dass der Code bereits vor 2 Monaten geändert wurde und der Kommentar vermutlich nicht mehr gültig ist. Eine Nachfrage beim Kollegen führt zu der Erkenntnis, dass der Kommentar überflüssig ist und es einfach vergessen wurde ihn zu entfernen.

Vor allem Produkte welche über mehrere Jahre hinweg von verschiedenen Personen entwickelt worden sind, leiden meist an ähnlichen Problem. Der Code ist sehr schwer zu Lesen. Die Folgen die durch diese Unlesbarkeit entstehen sind sehr weitreichend und dürfen nicht unterschätzt werden. In vielen Fällen hat unlesbarer Code schon zum „Tod“ von ganzen Programmen geführt, da die Weiterentwicklung unmöglich wurde.

Die CCD – Bewegung ist eine in den letzten Jahren immer stärker werdende Strömung die sich vor allem dafür einsetzt, Code so zu gestalten, dass die Aspekte Wartbarkeit, Lesbarkeit und die sich dadurch ergebende Zukunftssicherheit gegeben sind. Dadurch wurden bereits zahlreiche Publikationen, Artikel und auch Bücher veröffentlicht in denen diese Praktiken beschrieben werden. Es geht beim CCD in erster Linie um die Qualität des Codes hinsichtlich Lesbarkeit. Dieses Thema sollte in meiner Bachelorarbeit näher behandelt werden. Es wird dabei nicht auf das Thema Architektur eingegangen.

## Die Probleme moderner Software

Die Entwicklung von Software ist sehr ähnlich wie in der folgenden Grafik erläutert.

Dabei wird die Codebasis immer mehr erweitert und es werden immer weitere Features hinzugefügt. Oft werden auch Fehler behoben und dadurch verändert sich die Codebasis ebenfalls. An den meisten Softwareprojekten arbeiten auch zahlreiche verschiedene Entwickler und dadurch ergibt sich ein weiterer Faktor, der zur Unterschiedlichkeit der Codebasis beiträgt.

Aus diesen Voraussetzungen welche für nahezu jede Software, welche über einen längeren Zeitraum unterstützt und weiterentwickelt wird, zutreffen ergeben sich folgende Probleme:

* Es werden immer mehr Features hinzugefügt wodurch der Code immer mehr wird