

„Spiegeln, Spiegeln an der Wand, was ist das beste im ganzen Land“: Möglichkeiten und Erfahrungen des System-Benchmarking

Dr. Frank Simon (SQS)
(Frank.Simon@sqs.de)

Dr. Dieter Bolz (T-Mobile)
(dieter.bolz@t-mobile.de)

1 Problemstellung

Unter Code-Quality-Management (CQM) wird im Allgemeinen die Sicherstellung technischer Qualität in SW-Systemen verstanden. Da diese niemals perfekt sondern nur graduell vorliegt, ist eine Herausforderung bei der Identifikation technischer Risiken das Belegen deren *Signifikanz*. Bei Systemgrößen von mehreren Millionen Lines of Code und mehreren tausend Klassen, wie sie für die T-Mobile mit ihren hochkomplexen Abläufen und Systeminteraktionen notwendig sind, ist es nur selbstverständlich, dass jeder Qualitätsaspekt positive Befunde liefert. Beispiele solcher Aspekte sind z.B. Verletzungen des Information Hiding oder unvollständiges Exception-Handling. Werden diese Befunde mit Werkzeugen ermittelt, so verhindert der Output alleine aufgrund seines Umfangs jede effiziente Reengineering-Bearbeitung.

In diesem Papier wird ein Ansatz zur Lösung dieser Problemstellung vorgestellt, der auf Basis von Benchmarks einen direkten Vergleich mit einem Stand-der-Technik ermöglicht. Letzterer erlaubt erstmals konkrete Aussagen darüber, wie sich die Qualität eines Systems gegenüber einer Vielzahl anderer Systeme verhält und erlaubt damit eine sehr mächtige Priorisierung.

2 Lösungsansatz

Im Grimm Märchen *Dornröschen* erlaubt ein Spiegel das landesweite Benchmarking von Menschen bzgl. der Eigenschaft Schönheit: Auf Platz 1 liegt hierbei das Schneewittchen, gefolgt von der Stiefmutter auf Platz 2. Diesem Benchmarking müssen folgende Informationen zugrunde liegen:

- Ein gemeinsames Verständnis von Schönheit, das bis auf die Ebene konkret überprüfbarer /bewertbarer Merkmale herunter gebrochen ist.
- Ein Repository, in dem für jedes Merkmal die konkrete Ausprägung jedes Menschen des Landes abgelegt ist. Dies ist nötig, um ordinale Aussagen pro Merkmal machen zu können.
- Eine Aggregationsfunktion, die es erlaubt, die Vielzahl von Verfeinerungen der Eigenschaft Schönheit und deren individuelles Benchmarking gegenüber dem Repository evtl. gewichtet so zusammenzufassen, dass klare Aussagen der Art „*ist schöner als*“ möglich sind.

Unabhängig davon, ob diese Informationen bei Dornröschen tatsächlich existierten, so müssen sie zumindest für eine plausible und gebrauchstaugliche Portierung dieser Idee auf den Bereich der Software-Qualität bedacht werden:

- Für das gemeinsame Verständnis von technischer Qualität im T-Mobile-Kontext wurde mittels spezifischer bidirektionaler Q-Modelle für C++ und JAVA konkret formuliert, was unter technischer Qualität verstanden wird.
- Das Repository, das für ein Benchmarking der einzelnen Schönheitsfaktoren verwendet wurde, ist im Kontext des Forschungsprojektes QBench ([1]) entstanden und enthält anonymisiert Kennzahlen von über 130 Projekten.
- Die Aggregationsfunktion existiert bis heute nicht, da hierfür nicht nur die einzelnen Faktoren sondern auch deren Gewichtung allgemeingültig vorgegeben werden müssten. Diese Allgemeingültigkeit ist bereits bei Dornröschen bemerkenswert, schließlich scheint dort z.B. eine klare Gewichtung zwischen der Schönheit der Augen und der Haare zu existieren. In der Software-Technik reduziert sich das anvisierte Benchmarking daher auf ein Tupel von Einzelbenchmarks einzelner Faktoren.

2.1 Verwendete Meßinfrastruktur

Auf der Ebene der Code-Merkmale wurden Werkzeuge und Metriken verwendet, die heute bereits eine weite Verbreitung gefunden haben und insbesondere auch im Bereich des QBench-Projektes relevant sind. Typische Metriken sind Anzahl öffentlicher Methoden in Klassen oder Anzahl von Klassen- und Methodenzyklen.

Der Link zwischen den vorliegenden Metriken, Regeln und Vorgaben und den gewünschten Zielen in Form der gewünschten Qualitäts-Eigenschaften wurde im vorliegenden Fall durch 66 Qualitätsindikatoren für JAVA und 39 Qualitätsindikatoren für C++ ermöglicht.

2.2 Stand-der-Technik-Repository

Das im Kontext der T-Mobile erstellte Q-Modell kann als Untermenge des im Forschungsprojekt QBench erarbeiteten Q-Modells aufgefasst werden. Das dort verwendete Q-Modell wurde bereits für eine Vielzahl großer JAVA und C++-Projekte angewendet. Die einzelnen Datensätze des Repositories stammen hierbei aus Assessment-Projekten der SQS, des Fraunhofer-Instituts für Software-Technik und des Forschungszentrums Informatik an der Universität Karlsruhe.

Im Repository liegen die Risikodichten für eine Vielzahl von Qualitätsindikatoren anonymisiert vor. Die Normierung von absoluten Risikostellen bzgl. der Systemgröße auf Risikodichten ist hierbei wichtig, da anderenfalls ein größeres System

zwangsläufig immer eine schlechtere Qualität aufweisen wird, als ein kleines System.

2.3 Benchmarking

Eine sehr praxistaugliche Definition von Benchmarking findet sich in [2]. Demzufolge ist „Benchmarking [...] der methodische Vergleich von Prozessen und Produkten mittels Benchmarks von als besser identifizierten Vergleichspartnern. Die Vergleichspartner werden anhand von Ähnlichkeiten in der eigenen oder in anderen Organisationen gefunden. Ziel des Benchmarking ist es, die eigenen Prozesse und Produkte durch das Vorbild des Vergleichspartners entscheidend zu verbessern.“

Diese Definition taugt ebenfalls für das in dieser Arbeit beschriebene Vorgehen:

- Im vorliegenden Fall bezieht sich das Benchmarking auf Produkte in Form vorliegenden Quelltextes praxisrelevanter IT-Applikationen.
- Das Ziel des Benchmarks war für die T-Mobile deutlich die Verbesserung der entgegen dem Stand-der-Technik unterdurchschnittlich guten Qualitätsaspekte ihrer Systeme.
- Als Ähnlichkeit der Vergleichspartner wurde in jedem Fall die verwendete Programmiersprache verwendet, d.h. ein JAVA-System wurde mit JAVA-Systemen, ein C++-System mit C++-Systemen des Repositories verglichen.

3 Ergebnisse

In der folgenden Abbildung ist ein typischer Screenshot eines externen Benchmarkings dargestellt, wie er für 10 Projekte der T-Mobile erstellt wurde. Auf der X-Achse sind die verschiedenen standardisierten Qualitätsindikatoren aufgetragen. Ein Punkt im grünen Bereich (Intervall zwischen Minimum und unterem Quartil) bedeutet, daß das analysierte System in diesem Bereich zu den 25% der besten Projekte des Repositories gehört. Im Gegensatz dazu bedeutet ein Wert im roten Bereich (Intervall zwischen Maximum und oberem Quartil), daß das analysierte System bzgl. des betrachteten Aspektes zu den 25 der schlechtesten Systeme gehört.

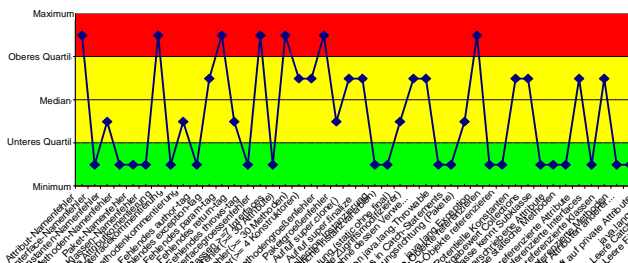


Abbildung 1: Beispiel eines externen Benchmarkings

Eine derartige Darstellung des externen Benchmarkings demonstriert sehr eindrucksvoll, bzgl. welcher Qualitätsaspekte das analysierte System den größten Handlungsdruck aufweist (im vorliegenden Beispiel sind das z.B. die 8 Qualitätsindikatoren im

roten Intervall) und wurde auch in den Projekten sehr effektiv eingesetzt. Insbesondere die Relativierung von i.d.R. absolut formulierten Programmiervorgaben (z.B. „kapsel alle Daten“) durch das Benchmarking mit einem Stand-der-Technik wurde sehr positiv für die Reengineering-Planung aufgenommen.

4 Erfahrungen

CQM ist ein mächtiges Instrument, um die technische Zukunftsfähigkeit großer IT-Systeme nachhaltig zu verbessern. Die punktuellen Einzelarbeiten, die jedes größere Projekt in diesem Bereich im Vorfeld bereits durchgeführt hatte, wurden durch das Benchmarking vervollständigt und untereinander vergleichbar. Durch den externen Benchmark mit einem Stand-der-Technik-Repository ist es einem Systemverantwortlichen erstmals möglich, die qualitative Ausprägung eines Systems mit dem Industrieschnitt zu vergleichen, um eine systematische Priorisierung gefundener Risiken für ein Reengineering vornehmen zu können. Diese Positionsbestimmung wurde von allen Seiten sehr neugierig aufgenommen und hat die Lebbarkeit von ansonsten eher akademisch anmutenden Vorgaben deutlich erhöht, da i.d.R. nicht auf eine Risikodichte von 0 sondern gegenüber der Standard-Risikodichte üblicher Systeme verglichen wird, die i.d.R. deutlich höher als 0 liegt (das CQM-Repository kennt z.B. kein einziges System, das sich vollständig und ohne Verletzung an die Forderung nach Datenkapselung hält).

Die T-Mobile wird diese Form der Qualitätssicherung in Zukunft verstärkt einsetzen, da sich deutliche Mehrwerte für die Projekte gezeigt haben; aufgrund der Werkzeugunterstützung stehen dem erreichten Nutzen insbesondere vergleichsweise geringe Kosten entgegen.

Ein nächster Schritt für die T-Mobile wird die Einbeziehung von X-Shoring-Systemen sein, um sicherzustellen, daß deren Qualität der hohen Qualität der Inhouse-erstellten Systeme der T-Mobile entspricht. Wenn sich der Bereich weiter etabliert hat, wird auch ein T-Mobile-weites, internes Benchmarking politisch gewünscht sein, so daß zwischen den Anwendungen bzgl. der technischen Güte ein Wettbewerb entstehen kann. Ein derartiges internes Benchmarking hilft insbesondere dem höheren T-Mobile-Management bei der maximal effektiven Nutzung vorhandener Konsolidierungsbudgets, da die relativen Potentiale der Systeme untereinander durch das Benchmarking offenkundig wird.

5 Referenzen

- [1] Forschungsprojekt QBench, vgl. www.qbench.de
- [2] Deutsches Benchmarking Zentrum: http://www.benchmarkingforum.de/bench_info_1.htm.