

Fehlerhäufigkeiten in objektorientierten Systemen: Ergebnisse einer Online-Umfrage

Lars Borner, Universität Heidelberg, Arbeitsgruppe Software Systeme
Matthias Hamburg, PSI AG, Düsseldorf
Stefan Jungmayr, Teradyne Diagnostic Solutions, Ismaning

Abstract: Für einen effizienten Einsatz von Qualitätssicherungsmaßnahmen ist die Kenntnis über die häufigsten Fehlerursachen entscheidend. Dieser Artikel beschreibt die Ergebnisse einer Online-Umfrage zu Fehlerhäufigkeiten in objektorientierten Systemen. Die Umfrage wurde vom GI-Arbeitskreis „Testen objektorientierter Programme“ im Zeitraum vom 19. August bis 31. Oktober 2005 durchgeführt. Die Fragebögen wurden von 1219 Teilnehmern ausgefüllt.

1 Einleitung

Die Kenntnis der häufigsten Fehlerursachen in einem gegebenen Testkontext ist für die Wahl von effizienten Qualitätssicherungsmaßnahmen entscheidend. Sind z.B. Fehler im Kontrollfluss eine häufige Fehlerkategorie, so sollte ein Testverfahren wie die Anweisungs- und Zweigüberdeckung angewendet, treten Datenflussfehler sehr häufig auf, so sollte eine Datenflussanalyse durchgeführt werden [7].

Im GI-Arbeitskreis „Testen objektorientierter Programme“ [13] haben wir in der Literatur nach Listen bzw. Statistiken von Fehlerhäufigkeiten gesucht und solche u.a. in [1], [2], [4], [5] und [6] gefunden.

Diese Literatur-Suche hat nur unzureichende Daten über Fehlerhäufigkeiten hervorgefördert. Darüber hinaus deckten die darin untersuchten Systeme und Fehlerkategorien objektorientierte Software nicht ausreichend ab. Daraus wurde die Idee zu einer Online-Umfrage unter Testern und Software-Entwicklern geboren.

Der Fokus der Umfrage lag auf Fehlern im Source-Code. Fehler in der Dokumentation oder im Entwicklungsprozess wurden bewusst in der Umfrage nicht berücksichtigt. Ebenso wurden Fehler bezüglich nicht-funktionaler Qualitätskriterien wie Benutzbarkeit, Wartbarkeit, Erweiterbarkeit und Zuverlässigkeit nicht betrachtet.

Die folgenden Kapitel beschreiben die Umfrage und geben eine erste Auswertung der Ergebnisse wieder. Eine detaillierte Beschreibung der Umfragedaten sowie der Auswertung ist in [3] enthalten.

2 Fragebogen

Der Online-Fragebogen wurde in zwei Teile gegliedert. Im ersten Teil wurden die Teilnehmer- und Projektdaten sowie die Häufigkeiten der Fehler-Hauptkategorien abgefragt. Dabei umfassten die Teilnehmerdaten die Test Erfahrung und die Position der Teilnehmer. Die Informati-

onen zu den verschiedenen Projekten beschränkten sich auf die eingesetzte Programmiersprache und die Branche, für die Software entwickelt und getestet wurde. Bei der Bewertung der Häufigkeiten der Fehlerkategorien standen den Teilnehmern im ersten Teil folgende sieben Fehler-Hauptkategorien zur Auswahl:

- Fehlerhafte Interpretation der Anforderungen,
- Fehler im Kontrollfluss,
- Fehler in der Berechnung,
- Fehler in Klassen oder Datentypen,
- Fehler im Datenfluss/Objektzugriff,
- Schnittstellenfehler,
- Konfigurationsfehler.

Die Häufigkeit der Fehlerkategorien wurde anhand der folgenden Skala bewertet: „nie“, „selten“, „oft“, „sehr oft“ und „keine Angabe“. Als Hilfe zu dieser Skala wurden Intervalle zur relativen Häufigkeit in Prozent angegeben [3].

Neben der Bewertung der Häufigkeiten zu den einzelnen Fehler-Hauptkategorien hatten die Teilnehmer im ersten Teil noch die Möglichkeit, eine eigene Fehler-Hauptkategorie und deren Häufigkeit anzugeben.

Im zweiten Teil des Fragebogens wurden zu den sieben Fehler-Hauptkategorien jeweils 4 bis 10 Fehler-Unterkategorien vorgegeben, die von den Teilnehmern nach ihrer Häufigkeit bewertet werden konnten. Damit die Teilnehmer nicht die Häufigkeit für alle 40 Fehler-Unterkategorien bewerten mussten, wurden ihnen nur die Dialoge der Fehler-Unterkategorien zu den am höchsten (mit „oft“ oder „sehr oft“) bewerteten Hauptkategorien dargestellt. Somit wurde jede einzelne Unterkategorie durchschnittlich von ca. der Hälfte der Teilnehmer bewertet.

Die Definition der Fehlerkategorien orientierte sich an den Literatur-Angaben in [1], [2] und [4] sowie an den persönlichen Erfahrungen der Arbeitskreis-Mitarbeiter. Der Online-Fragebogen wurde von Lars Borner mit Servlets realisiert und am Lehrstuhl „Software-Systeme“ der Universität Heidelberg gehostet [14].

3 Durchführung der Umfrage

Die Umfrage wurde durch die Zeitschrift OBJEKTSpektrum [11], die Computer-Zeitung [8] sowie einen News-Ticker in heise.de [9] angekündigt. 1219 Teilnehmer hatten vom 19. August bis zum 31. Oktober 2005 die Möglichkeit genutzt, an der Umfrage teilzunehmen. (Die Teilnahmebedingungen sind in [3] wiedergegeben.)

Als Anreiz für die Teilnahme an der Umfrage wurden freundlicherweise vier Test-Fachbücher als Gewinne durch den Verlag dpunkt zur Verfügung gestellt.

Die öffentliche Ziehung der Gewinner erfolgte beim 23. Treffen der GI-Fachgruppe TAV [12] am 17. und 18. November 2005 in München sowie beim Seminar „Effiziente Testmethoden“ am 13.12.2005 an der Universität Heidelberg¹. Die Gewinner der Umfrage waren Thomas Frey (Frankfurt a.M.), Michel Fügeisen (München), Michael Gerlinger (Filderstadt-Bernhausen) und Andreas Krauß (Altlußheim).

4 Auswertung der Teilnehmer- und Projektdaten

Die meisten Teilnehmer der Umfrage arbeiten für Software-Firmen (44%), gefolgt von Dienstleistungsunternehmen (29%) und Unternehmen im technischen Bereich (16%). Die Mehrheit (61%) der Teilnehmer hatte eine Testerfahrung zwischen einem und fünf Jahren angegeben. Knapp drei Viertel (73%) der Teilnehmer sind als Entwickler bzw. Architekt in ihrem Umfeld tätig. Nur ein sehr kleiner Prozentsatz von 2,5% bezeichnet sich als Tester. Die in den Projekten der Teilnehmer am häufigsten eingesetzten Programmiersprachen sind Java (41%) und C++ mit (28%).

5 Ergebnisse

Die folgende Liste beschreibt die wichtigsten Resultate der Umfrage. Dabei werden die Fehler-Hauptkategorien in absteigender Anzahl der Häufigkeitsangabe „sehr oft“ genannt:

1. *Fehlerhafte Interpretation von Anforderungen*: Die

fehlende Implementierung von impliziten Anforderungen sowie falsch verstandene Kundenanforderungen sind hier die häufigsten Fehler-Unterkategorien.

2. *Konfigurationsfehler*: Das Fehlen der aufgerufenen Komponente bzw. das Vorliegen einer falschen Version wurde hier als häufigste Fehlerursache genannt.
3. *Fehler im Datenfluss bzw. Objektzugriff*: Diese Fehler betreffen am häufigsten Datenflussanomalien.
4. *Schnittstellenfehler*: Bei diesen steht die Nichtbeachtung von Kontrakten (d.h. die Verletzung von Vor- und Nachbedingungen bzw. die Nichtbeachtung von Seiteneffekten) im Vordergrund.
5. *Fehler im Kontrollfluss*: Hierbei sind Fehler in der Behandlung von internen Ausnahmefällen (Exceptions) und in parallelen Prozessen am häufigsten genannt worden.
6. *Berechnungsfehler*: Die häufigste Fehler-Unterkategorie bezieht sich auf die Ressourcenverwendung von z.B. temporären Daten oder das Speicher-Management.
7. *Fehler in Klassen oder Datentypen*: Diese treten insgesamt am seltensten auf, betreffen dann aber am häufigsten die Typtransformation.

Bemerkenswert ist, dass Konfigurationsfehler an zweiter Stelle genannt wurden, noch vor den traditionellen Fehlerquellen wie Fehler im Datenfluss bzw. Objektzugriff oder Schnittstellenfehler. Fehler im Kontrollfluss und in der Berechnung, die insbesondere in der prozeduralen Entwicklung die größte Aufmerksamkeit durch Testmethoden erfahren haben, lagen in der Bedeutung stark zurück.

Die Auswertung der benutzerdefinierten Fehlerkategorien hatte keine neuen nennenswerten Fehlerkategorien

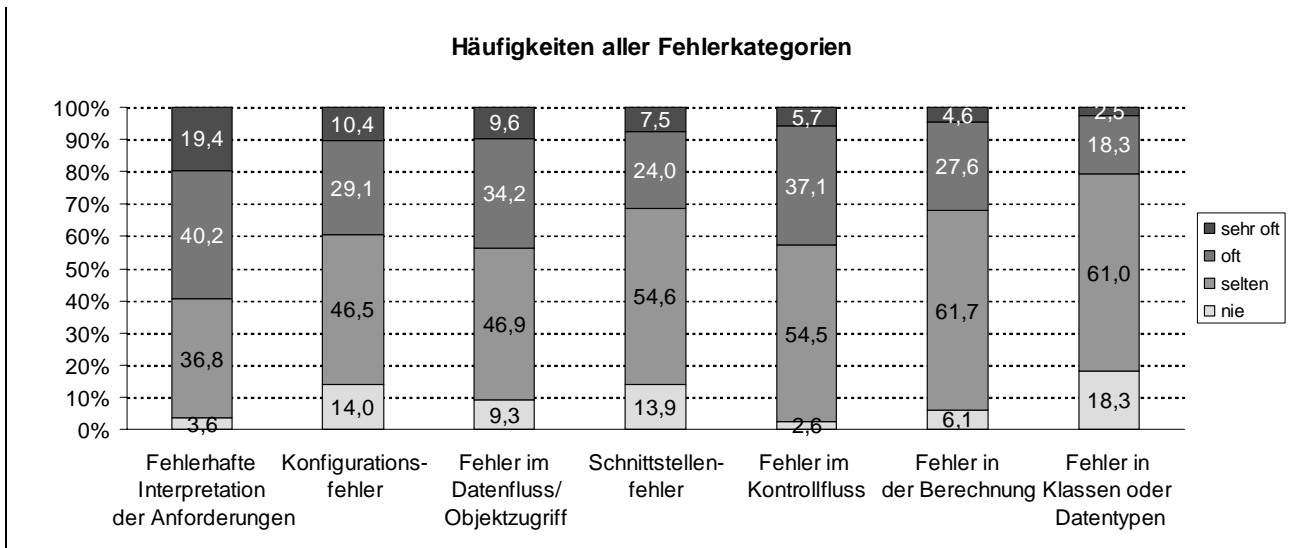


Abbildung 1: Häufigkeit der Fehlerkategorien

¹ Eine zweite Ziehung wurde notwendig, weil zwei der vier in München gezogenen Gewinner keine gültige Email-Adresse als Kontaktmöglichkeit angegeben hatten.

zur Korrektheit des Codes aufgedeckt. Dies deutet darauf hin, dass die von den Autoren vorgegebene Menge an Fehlerkategorien die potentiell in der Praxis auftretenden Fehler weitestgehend abdeckt.

6 Fazit

Die Definition der Fehlerkategorien, die Implementierung des Online-Fragebogens, die Benachrichtigung der Gewinner sowie die Auswertung der Daten haben wesentlich mehr Aufwand verursacht, als ursprünglich angenommen. Die große Anzahl der Teilnehmer und das Interesse innerhalb der Entwicklungsgemeinde hat dies jedoch mehr als entschädigt. So wurden die Ergebnisse auf Anfrage bei der iX-Konferenz „Bessere Software“, 22.-23. November 2005, in Köln in Form eines Posters präsentiert [10]. Zusätzliche Veröffentlichungen der Ergebnisse und eine weitergehende statistische Auswertung der Umfrage-Daten sind geplant.

Einladung: Wenn Sie die Ergebnisse der Umfrage mit Mitarbeitern des Arbeitskreises TOOP diskutieren bzw. eigene Erfahrungen, Kommentare oder Zusatzinformationen einbringen wollen, so laden wir Sie herzlich ein, über die Homepage des Arbeitskreises TOOP [13] Kontakt mit den Autoren aufzunehmen

7 Dank

An der Entwicklung der Umfrage waren Michael Averstegge, Falk Fraikin, Kay Krämer, Roger Müller, Andreas Schönknecht und Mario Winter beteiligt. Weiters gilt unser Dank den in [3] genannten Unterstützern und Reviewern.

8 Referenzen

- [1] B. Beizer. *Software Testing Techniques*, 2. Ed. International Thomson Computer Press, 1990.
- [2] R. V. Binder. *Testing Object-Oriented Systems*. Addison-Wesley, 2000.
- [3] L. Borner, F. Fraikin, M. Hamburg, S. Jungmayr, A. Schönknecht. *Fehlerhäufigkeiten in objektorientierten Systemen: Basisauswertung einer Online-Umfrage*. Technical Report SWEHD-TR-2006-01, Universität Heidelberg, Institut für Informatik, Februar 2006. URL: <http://www-swe.informatik.uni-heidelberg.de/research/publications/reports.htm>
- [4] N. E. Fenton, S. Lawrence Pfleeger: *Software Metrics 2*. Ed. International Thomson Computer Press, 1997
- [5] G. Myers. *The Art of Software Testing*, 2. Ed. John Wiley & Sons, 2004.
- [6] M. Roper. *Software Testing*. McGraw-Hill, 1994.
- [7] A. Spillner, T. Linz: *Basiswissen Softwaretest*. 3. überarbeitete Auflage, dpunkt.verlag, Juli 2005.
- [8] <http://www.computerzeitung.de>
- [9] <http://www.heise.de/>
- [10] <http://www.ix-konferenz.de/einstieg.php?konferenzid=5>
- [11] <http://www.sigs-datacom.de/sd/publications/os/index.htm>
- [12] <http://www.gm.fh-koeln.de/~winter/tav/>
- [13] <http://toop.gi-ev.de>
- [14] http://mars.informatik.uni-heidelberg.de:8080/Questionnaire_TooP/TooP/Questionnaire