

Cut-And-Paste Programming

Antipattern

Tim Kleinschmidt, Lukas Kohler

Agenda

- Hintergrund
- Ausprägung
- Konsequenzen
- Typische Gründe
- Ausnahme

Hintergrund

- Sehr gebräuchlich, aber oft missbraucht
- Existierender Code wird modifiziert, statt von Grund auf neu zu programmieren
- Kann positive Effekte haben
- Verschlechtert die Codequalität bei übermäßigem gebrauch

Ausprägung

- Wiederholung gleicher Codesegmente in Softwareprojekt
- Viele unerfahrene Entwickler am selben Projekt lernen aus Codebeispielen der erfahrenen Entwickler
- Anpassung der Beispiele auf eigene Anforderungen
- Codeduplizierung wirkt sich kurzzeitig positiv auf „Line-Count“-Metrik aus
- Leicht erweiterbar, da Kontrolle bei Entwickler liegt

Konsequenzen

- Selber Bug bleibt bestehen, trotz Behebung an verschiedenen Stellen
- Beheben des selben Bugs auf unterschiedliche Weise
- LoC steigen an ohne hinzufügen neuer Funktionalität
- Unnötige Erhöhung des Aufwands für Code-Reviews
- Wiederverwendbarkeit des Codes bei minimalem Aufwand
- Sprunghafter Anstieg der Softwarewartungskosten

Typische Gründe

- Zu großer Aufwand wiederverwendbaren Code zu produzieren
 - Schnelle Ergebnisse vor zu langem Entwickeln
- Bestehender wiederverwendbarer Code nicht ausreichend dokumentiert
- Fehlende Voraussicht bei Entwicklern
- Entwickler kopieren Beispiele um neue Technologien kennenzulernen

Ausnahme

- Akzeptabel wenn einziges Ziel die schnelle Fertigstellung des Codes ist
 - Nachteil ist der erhöhte Wartungsaufwand