



University
of Basel

Software Engineering - Eigenschaften

M. Lüthi, Universität Basel, 9. August 2022

Software - ... ein (noch) undurchschaubares von Menschen geschaffenes Wunder.

Anonyme Teilnehmer*in GymInf

Was erhoffe ich mir in der Vorlesung zu lernen?

.....
wie baut man eine App? Für SchülerInnen Auskunft zu geben selbst Apps zu bauen

📖 Ich möchte gerne die Theorie aus den Programmiervorlesungen in eine Praxis umsetzen, etwas Zusammenhängenderes betrachten, als nur Snippets

Ich brauche insgesamt 4 Wahlpflichtvorlesungen und hab erst 2... 😊 Ausserdem erhoffe ich mir, ein bisschen einen Übergang aus dem Programmieren zu den "fertigen Apps", das fehlt mir nämlich komplett... Ich kann ein bisschen Code schreiben, aber wie man dann zu etwas fertigem kommt...?

📖 Thema: Wie strukturiere ich grössere Softwareprojekte und die Zusammenarbeit im Team?

.....
Schnittstelle zwischen Programmieren/Scripting und der normalen Welt.

The mess we are in



- We've made a mess
- We need to achieve stability
- Quantum mechanics sets limits to the ultimate speed of computation
- We need that
- Identify nations and places
- Build the computers
- Massive low power computers - no net environmental damage

<https://www.youtube.com/watch?v=IKXe3HUG2I4>

Agenda 09. August 2022

-
- 1 Projekt: Auswählen/Definieren eines Kursprojekts (15')
 - 2 Eigenschaften und Qualitätsmerkmale von Software (60')
 - 3 Einführung Git und Github (30')
 - 4 Praktische Übung 2: Erste Änderungen anbringen (90')
-

Projektauswahl

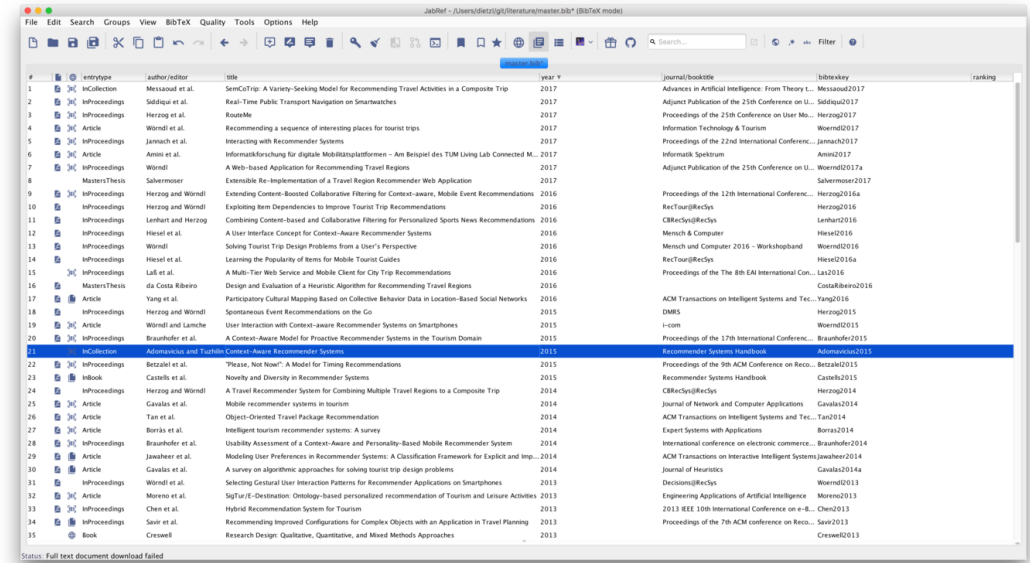
Informationen zum Projekt

Ziel

- In Gruppen (≥ 2 Personen) an einer Verbesserung von JabRef zu implementieren

Ablauf

- Es werden alle Projektphasen durchlaufen
 - Anforderungsanalyse
 - Prototyping / Design
 - Implementation
 - Testing
 - Dokumentation



Mögliche Projekte

- Nützliche oder lustige selbstdefinierte Erweiterungen von JabRef
- Bearbeitung eines Offenen Issues aus dem JabRef Repository

The screenshot shows the GitHub interface for the JabRef repository. The top navigation bar includes links for Code, Issues (317), Pull requests (25), Actions, Projects (6), Wiki, Security, and Insights. Below the navigation bar, there are filters for Labels (59) and Milestones (3), and a 'New issue' button. The main content area displays a list of 317 open issues, with 3,331 closed issues. The issues are sorted by 'Sort' and filtered by 'is:issue is:open'. The list of issues includes:

- Automatic field editor: 3 sentences not translated** (good first issue, internationalization) - #9014 opened 5 hours ago by mlep, 2 tasks done, v5.7
- Add support for also updating citationkeys in linked .tex files when updating citationkeys in the JabRef database** (external files, type: enhancement) - #8992 opened 11 days ago by ThiloteE
- Linked files with relative path not always found (unlike in older JabRef version)** (external files, type: bug) - #8991 opened 13 days ago by AEgit, 2 tasks done
- v5.7 Preferences that slow down selection of entries in large bibliographies** (preferences, type: performance) - #8984 opened 17 days ago by ThiloteE, 2 of 3 tasks
- The process of typing into fields is too slow in a large database** (entry-editor, type: bug, type: performance) - #8977 opened 19 days ago by ThiloteE, 1 task
- Deleting many entries in a large database takes way too long** (type: bug, type: performance) - #8976 opened 19 days ago by ThiloteE
- Enabling timestamps and copying and pasting many entries is very slow** (type: bug, type: performance) - #8975 opened 19 days ago by ThiloteE
- Activating a special field should not unselect an entry** (entry-editor, search, type: bug) - #8958 opened 26 days ago by mlep, 2 tasks done

Anforderungen der Erweiterungen müssen gut spezifizierbar sein.

- Simpler Bugfix nicht geeignet.

Einige mögliche offizielle Issues

Send to Kindle:	https://github.com/JabRef/jabref/issues/6186
Add right click menu for main table header	https://github.com/JabRef/jabref/issues/6601
Integrate check-bib-for-predatory	https://github.com/koppor/jabref/issues/348
Create a better solution for defining multiline fields	https://github.com/JabRef/jabref/issues/6448
Support Citation Style Language (CSL) Styles in LibreOffice/OpenOffice	https://github.com/JabRef/jabref/issues/8893
Make abbreviations also working for conferences	https://github.com/koppor/jabref/issues/360
Add Mathematics Subject Classification as tooltip to keywords	https://github.com/koppor/jabref/issues/423
Show information about the journal	https://github.com/JabRef/jabref/issues/6189
Add support of BibLaTeX extended name format	https://github.com/JabRef/jabref/issues/4558
Support for multi-paper PDFs (AKA proceedings)	https://github.com/JabRef/jabref/issues/8128
Install and check browser extension from within JabRef	https://github.com/JabRef/jabref/issues/8400
Synchronisation with Overleaf	https://github.com/JabRef/jabref/issues/156

Eigene Recherche nötig, ob Issue wirklich geeignet ist

Bewertung

- Vorbereitende Übungen (20%)
- Pflichtenheft (20%)
- Signdokument (20%)
- Testplan (10%)
- Praktische Umsetzung (30%)

Bewertung pro Kriterium: 0 (ungenügend), 1 (gut) oder 2 (sehr gut)

Hausaufgabe (bis Freitag, 12. August)

- Funktionalität von JabRef explorieren
- Gruppen bilden
- Projekt aussuchen
- Kurzpräsentation erstellen (2-5 Minuten)

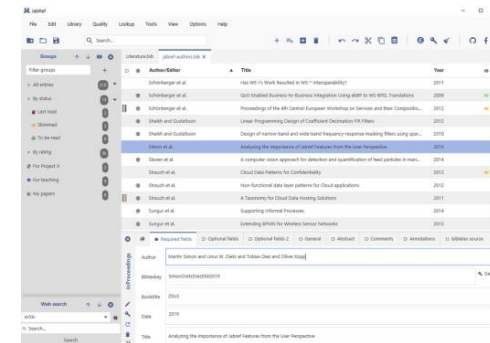
Präsentation

Es sollten folgende Fragen angesprochen werden:

- Was macht unsere Erweiterung
 - Welche Hauptschwierigkeit erwarten wir
 - Schätzung des Projektaufwands (unmöglich, aber machts trotzdem)
 - Mögliche Aufteilung der Arbeiten in Projektgruppe
-

Eigenschaften von Software

Das Produkt “Software”

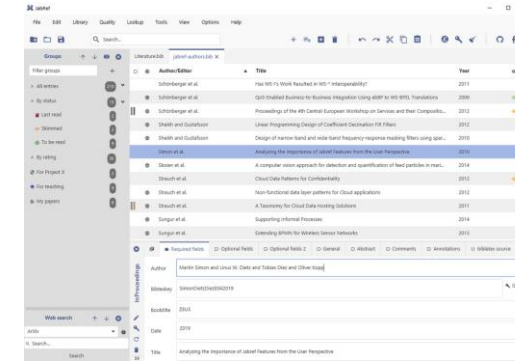


Software ist *irgendwie* anders als andere Produkte? Aber wie?

Eigenschaft 1: Formbarkeit

Software ist formbar (engl. malleable)

- Das Produkt lässt sich einfach verändern
- Keine teure Änderung der Produktionsanlagen nötig



Eigenschaft 1: Formbarkeit

Software ist Formbar – aber das Design ist starr

Software wird verändert ohne das Design anzupassen

- Produkt und Design divergieren



Eigenschaft 2: Immaterialität

Traditionelles Produkt

- Herstellung ist grösster Kostenfaktor
- Fokus: Reibungslose Produktion garantieren

Qualitätssicherung in Produktion

Software

- Herstellung ist trivial (kopieren)
- Fokus: Design und Implementation

Qualitätssicherung in Entwicklungsprozess

Eigenschaft 3: Software untersteht nicht den Naturgesetzen

Keine/Wenige vorderfinierten Regeln und verlässlichen Strukturen

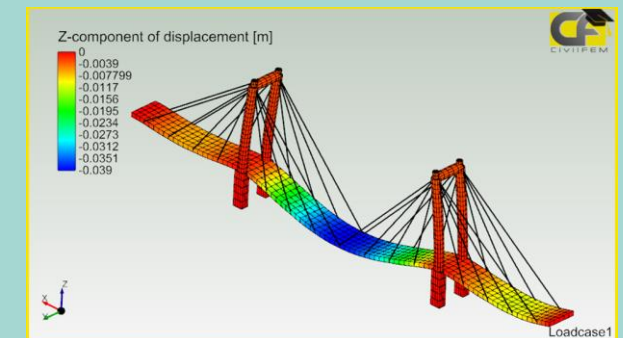
Logisch Folgern / Ableiten von Eigenschaften ist schwierig

- Erschwert Entwicklung und Verifikation

Alle logischen Strukturen und Regeln müssen explizit designed und dokumentiert werden.

Physik als Einschränkung in traditionellem Engineering

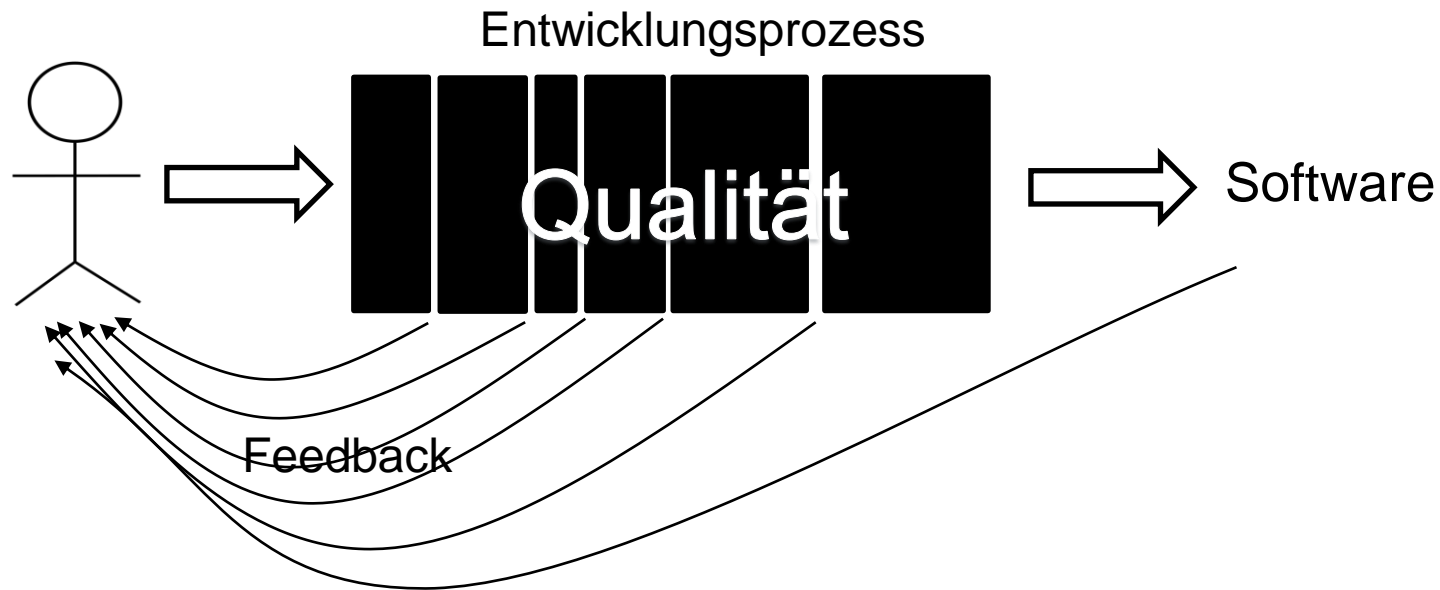
- definiert Lösungsraum
- Struktur in Form von Gesetzen (Mathematik)
- hilft uns Anforderungen zu testen



Eigenschaften: Überblick

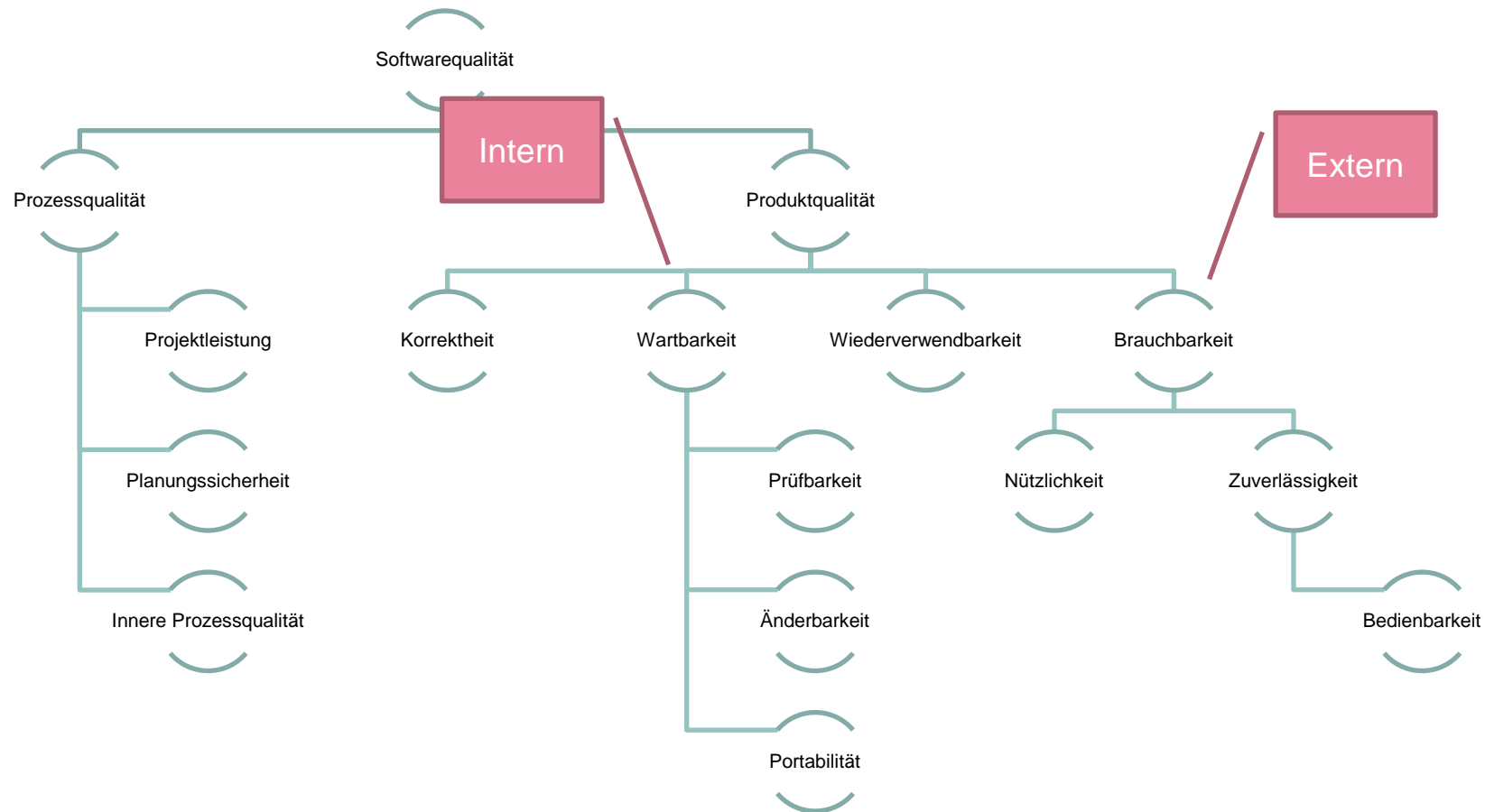
- Software lässt sich leicht ändern - das Design aber nicht
- Die Qualität der Software wird im Entwicklungsprozess erreicht
- Verlässliche Strukturen kommen nicht von aussen, sondern müssen explizit designed werden

Softwareentwicklung braucht Disziplin und gute Prozesse.



Qualitätsmerkmale von Software

Taxonomie



Interne vs. Externe Qualitätsmerkmale

- **Intern:** Sieht das Entwicklungsteam
- **Extern:** Sieht Benutzer:in

Beispiel 1: Korrektheit

Software ist korrekt, wenn sie die spezifizierten funktionalen Anforderungen erfüllt

- Erfordert spezifizierte Anforderungen
 - Korrektheit ist absolut
 - Programm ist nicht "ein bisschen falsch"
 - Formale Spezifikationen erlauben Korrektheitsbeweise
-

Beispiel 2: Zuverlässigkeit

Wahrscheinlichkeit, dass in einem bestimmten Zeitintervall kein Fehler auftritt

Intuitiv: Der Benutzer kann sich auf die Software verlassen



Beispiel: Zuverlässig / Korrekt

Ist dieses Programm korrekt?

Ist es zuverlässig?

```
double plus(double a, double b) {  
    double sum = a + b;  
    Random r = new Random();  
    if (Math.abs(r.nextGaussian()) < 1000) {  
        return sum;  
    } else {  
        return sum + 1;  
    }  
}
```

Robustheit

Software verhält sich auch unter unvorhergesehenen Umständen sinnvoll

Beispiele:

- Hardwareausfall
- Netzausfall
- Falscher Input

Schwierig formal zu spezifizieren

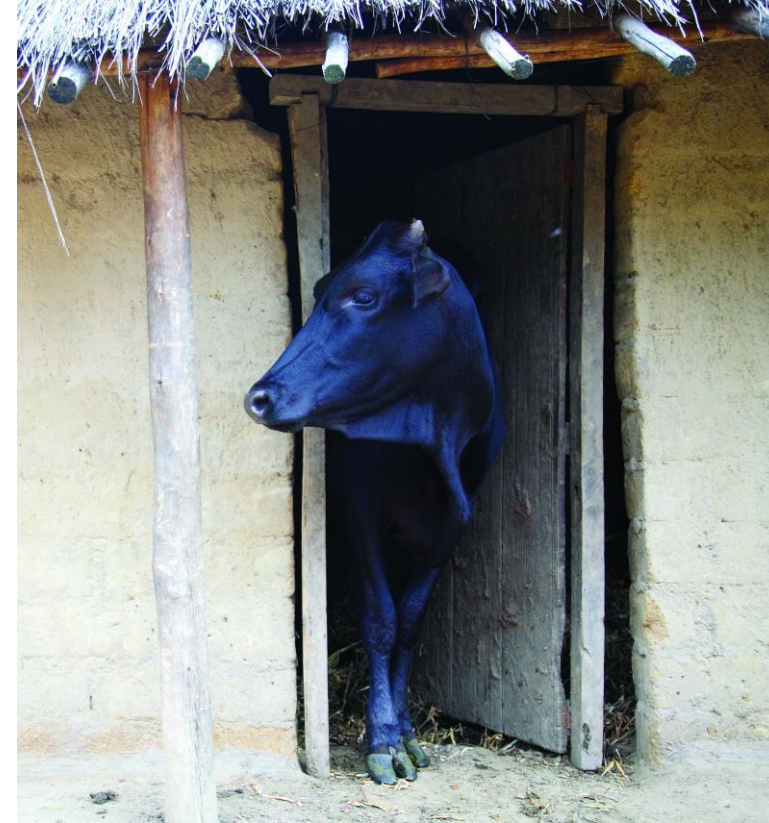


Korrektheit versus Robustheit
Anforderungen spezifiziert → Korrektheit
Anforderung nicht spezifiziert → Robustheit

Wartbarkeit



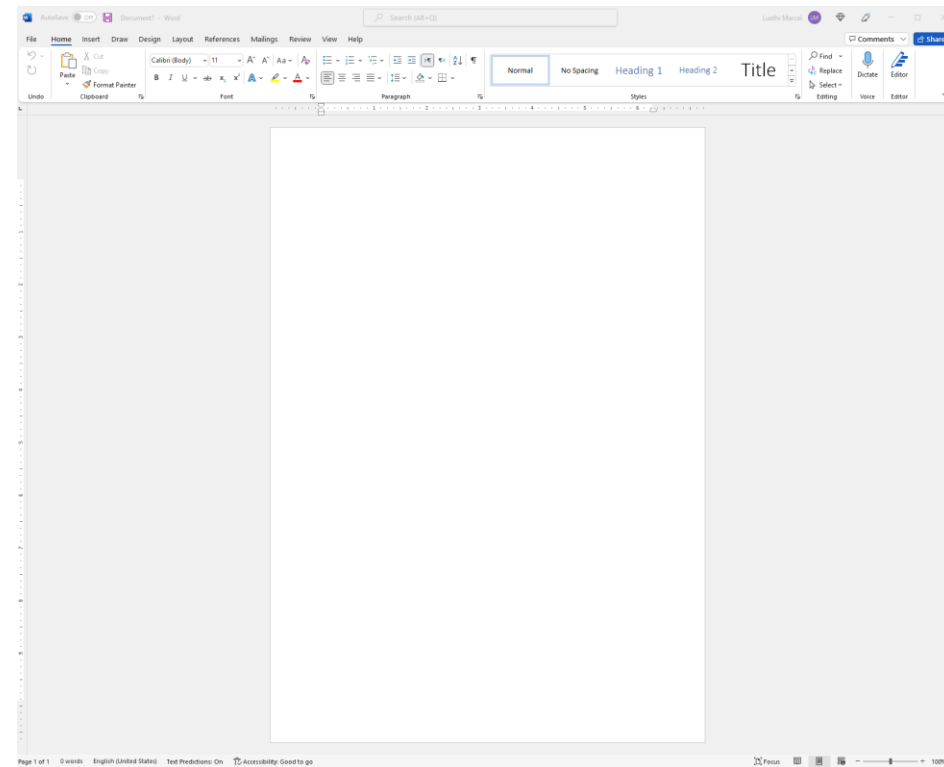
Software ist nicht das



sondern das!

Wartbarkeit

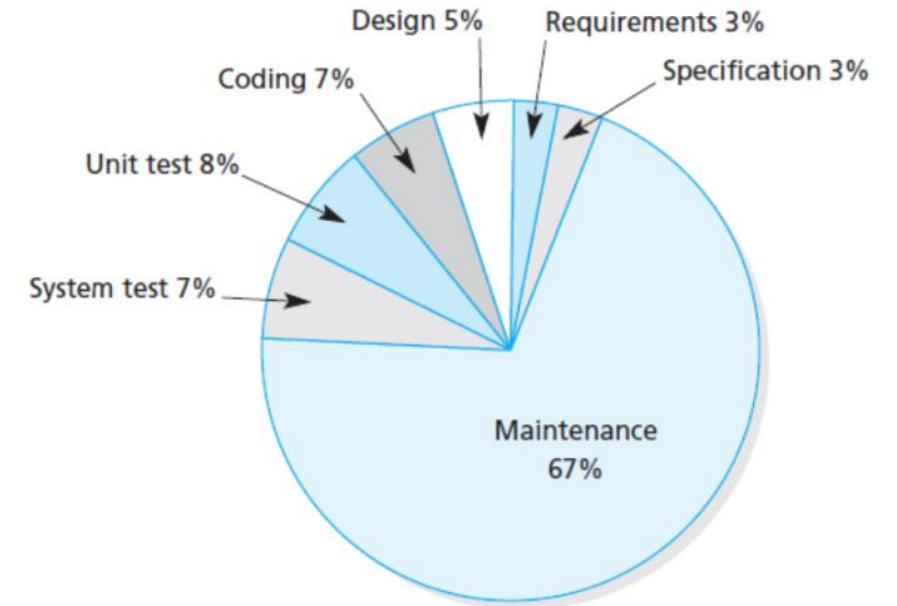
Wie einfach ist es die Software weiterzuentwickeln und an neue Gegebenheiten anzupassen?



Wartbarkeit

Gründe von Software-Änderung:

- Neue Hardware
- Neue Betriebssysteme
- Neue Anforderungen an Software
 - Gesetze
 - Marktdruck
- Fehlerbehebung
- Verbesserung (Performance / Wartbarkeit, ...)



Quelle: Software Engineering for Students, D. Bell

Grossteil der Kosten fällt nach der Entwicklung an!

Verständlichkeit

Wie einfach ist es den Code zu verstehen?

```
#include
int l;int main(int o,char **O,
int I){char c,*D=O[1];if(o>0){
for(l=0;D[l]
];D[l
++]-=10){D[l++]-=120;D[l]-=
110;while(!main(0,O,l))D[l]
+=20;putchar((D[l]+1032)
/20);}putchar(10);}else{
c=o+(D[I]+82)%10-(I>l/2)*
(D[I-l+I]+72)/10-9;D[I]+=I<0?0
:!(o=main(c/10,O,I-1))*((c+999)
)%10-(D[I]+92)%10);}return o;}
```

Winner 2001, Best Short Program, International Obfuscated C code Contest

Wiederverwendbarkeit

Wie einfach ist es die Software in anderem Kontext wiederzuverwenden?

- Komponentenbasierte Softwareentwicklung
 - Kosten für Verifikation, Wartung, etc. pro Komponente und nicht pro Produkt

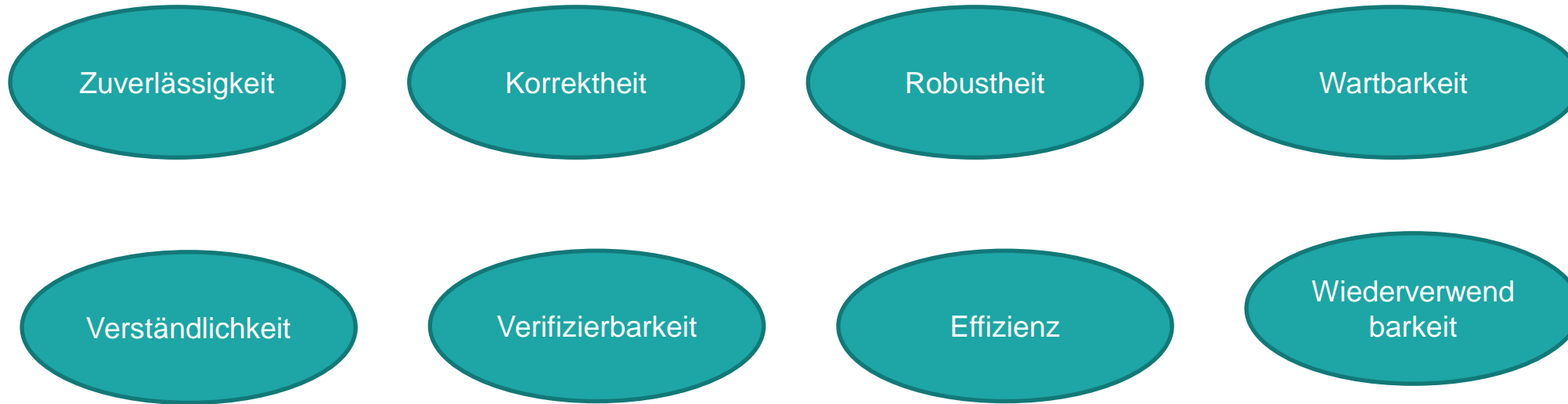
Portabilität

Grad der Plattformunabhängigkeit einer Software

Plattformen:

- Verschiedene Betriebssysteme
 - Verschiedene Browser
 - Verschiedene Hardware
-

Qualitätsmerkmale von Software



- Zeichnen Sie einen gerichteten Graphen, wobei es jeweils eine Kante gibt, wenn ein Merkmal ein anderes beeinflusst
 - Was können Sie vom resultierenden Graphen ableiten?
 - Gibt es Zyklen im Graphen? Was bedeuten diese?
-

Zusammenfassung

- Korrektheit von Software ist immer durch Anforderungen bestimmt.
 - Ohne Anforderungen gibt es keine Korrektheit, nur Robustheit
 - Gute Software muss wartbar sein.
 - Ein Grossteil der Kosten wird durch Wartung verursacht
 - Das Qualitätskriterium Wartbarkeit wird durch viele andere Qualitätskriterien beeinflusst.
-