**Delphi method test cost estimation:**

Analysis of the Future (Delphi Method) <http://www.dtic.mil/docs/citations/AD0649640> (Olai Helmer, 1967)

* Erklärung der Delphi Technik allgemein:
  + Mehrstufige Befragungen mit Feedback
  + Verkleinert range der antworten
  + Erlaubt aussage eines gruppen konsensus

An Estimation Model for Test Execution Effort (<https://ieeexplore.ieee.org/abstract/document/4343738/>)

* Test Cost Estimation to run Test Suites! (not, too applicable here)

# Standard für automatisierte Function-Point-Analyse

<https://m.heise.de/developer/artikel/Standard-fuer-automatisierte-Function-Point-Analyse-2067044.html>

* Src-Code relevante Messgröße
  + Fachwissen wird nicht berücksichtigt
* Zu identifizierende Artefakte:
  + Anwendungsdaten
  + Logik der Applikation über alle Schichten
* Bisherige Automatisierte Software zählt:
  + Gewichtete Transaktionen und ihre Komponenten
    - Data element types
    - File types referenced
    - Record element types
  + Außerdem: Messung des Umfangs von Änderungen
    - Vorrausetzung: „Baselining“ (analyse im release vor erfolgter änderung)
* Mögliche Schlussfolgerungen einer AFPA:
  + Messung der Qualität und Komplexität
  + Identifizierung ineffizienter Prozesse
  + Messung der Effekte von Optimierungsmaßnahmen.

**Function Point Measurement Tool for UML Design Specification**

**Keyword: Zyklomatische Komplexität:**

**(McCabe-Metrik)**

“Linear unabhängige Verzweigungen in einem Kontrollfluss“ (Wikipedia)

### Berechnung durch Anzahl Binärverzweigungen:

Komplexitätsmaß M = Anzahl binären Verzweigungen plus 1.

* *M* is an upper bound for the number of test cases that are necessary to achieve a complete [branch coverage](https://en.wikipedia.org/wiki/Branch_coverage).

Bei mehreren Kontrollflussgraphen (p > 1): M = b + p

b: Anz. Binärverzweigungen, falls mehr als 2 Zweige: b= Anz. Zweige – 1

p: Anz. Einzelner Kontrollflussgraphen (1 Graph pro Funktion)

**Alternativ:**

M = E – N + 2P

E: Anz. Kanten in Graph

N Anz. Knoten in Graph

P: Anz. Verbundener Komponenten

## Interpretation der Metrik

Laut McCab Zahl nicht höher als 10, sonst zu komplex

Zyklomatische Zahl erhöht sich nur dann, wenn wenn verzweigende Anweisungen wie IF eingefügt werden,

Es kann also lediglich eine Aussage über den Testaufwand (Anzahl der zu testenden unabhängigen Programmpfade) getroffen werden.

**Eignet sich gut zur Umsetzung**

**Frage**: Wie weiße ich Komplexität einer Funktion, die sich über mehreren Planblattseiten erstreckt, einer Planblattseite zu.

Allen, die dazugehören, nur der letzten, aufsteigend zur letzten gewichtet. Etc

Bei Implementierung:: Halting Problem Encounter

.

**Keyword: Scholar: Cyclomatic Complexity**

<https://ieeexplore.ieee.org/abstract/document/1702388/> - A complexity measure

<https://dl.acm.org/citation.cfm?id=48322> - critique of cyclomatic complexity as a software metric

**Keyword: Automated Function Point Counting**

* Quite expensive to develop; not as easy as Halstead or McCabe

<https://nesma.org/forums/topic/automated-function-point-counting/>

**Keyword: cyclomatic complexity in model-based software**

<http://www.engpaper.com/complexity-analysis-of%C2%A0simulink-models-to-improve-the-quality-of-outsourcing-in-an-automotive-company.htm>

<http://tucs.fi/publications/view/?pub_id=tOl11a>