

COBOL-Programm HPROG1

Redokumentation

Erstellt von: pro et con Innovative Informatikanwendungen GmbH
Reichenhainer Straße 29a
09126 Chemnitz

Erreichbar unter: 0371/270951-0

Stand: 04.09.2018



Inhaltsverzeichnis

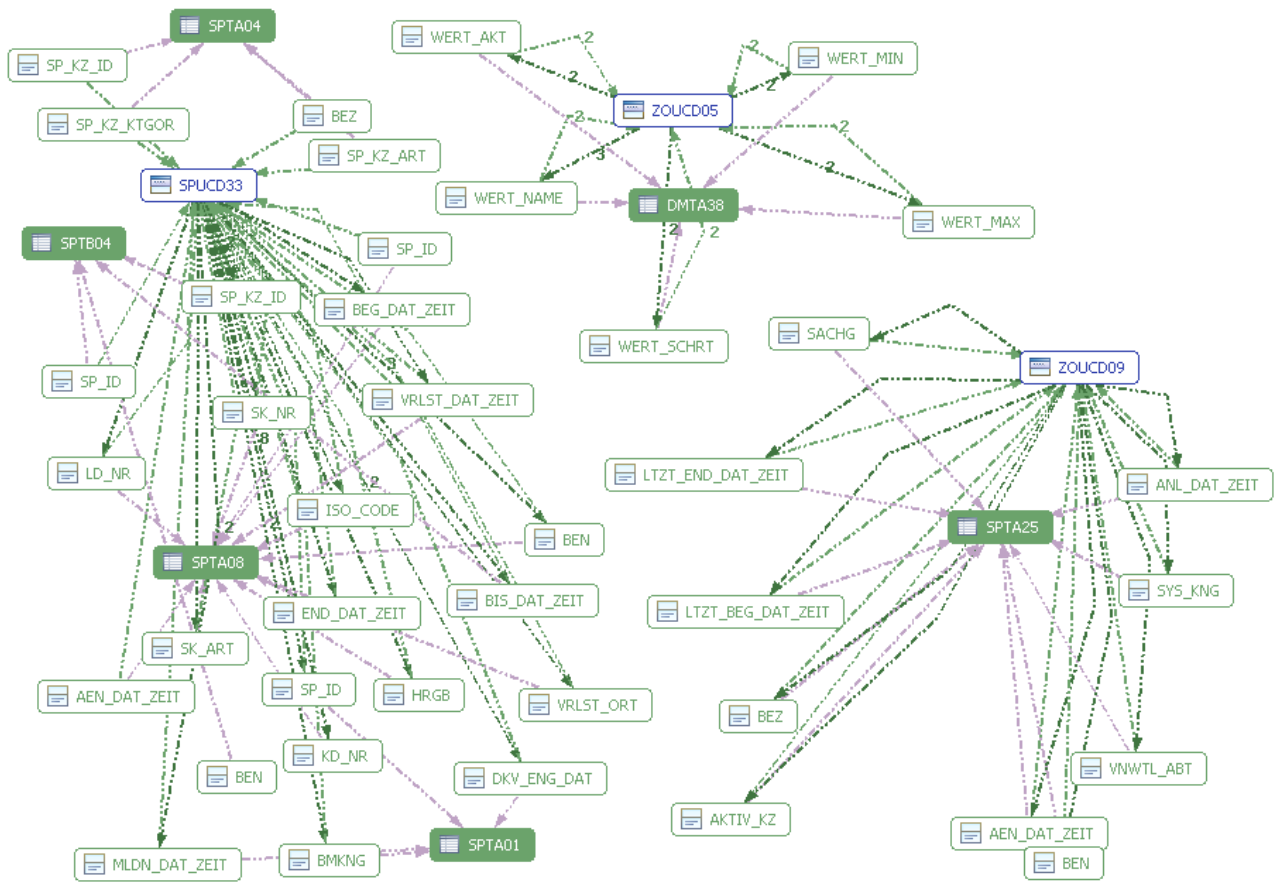
1	Globale Analyse	3
2	Applikationswissen	4
2.1	Aufrufhierarchie	4
2.2	Nutzung von Datenbanktabellen	4
2.3	Nutzung von Datenfiles	5
2.4	Metriken	6
2.4.1	Metrik nach Halstead	6
2.4.2	Metrik nach McCabe	8
2.4.3	Metrik Lines of Code	9
3	Feingranulare Analyse des Programm HPROG1	11
3.1	Programmstatistik	11
3.2	Metriken	12
3.3	Genutzte Datenfiles	12
3.4	Genutzte Datenbanktabellen	13
3.5	Datenfluss eines ausgewählten Objekts	13
3.6	Steuerfluss eines ausgewählten Paragraphen	14
3.7	Aufrufhierarchie	15



1 Globale Analyse

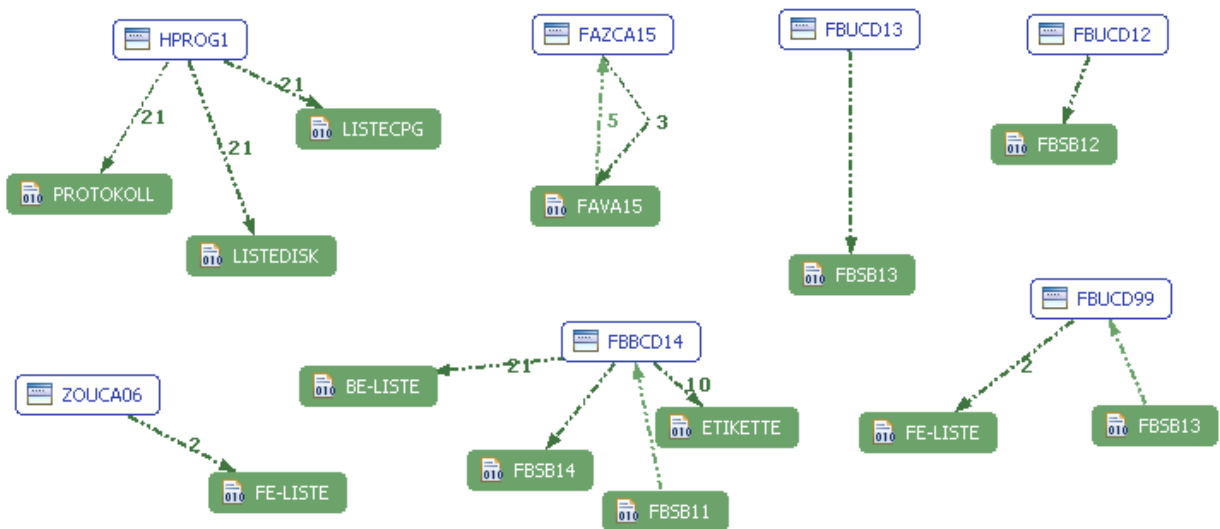
Es wurde folgendes Mengengerüst ermittelt:

Art	Anzahl
Hauptprogramme	4
Unterprogramme	24
Fehlende Unterprogramme	25
Copybooks	39
Fehlende Copybooks	10
Datenfiles	13
SQL-Tabellen	6



2.3 Nutzung von Datenfiles

Die Grafik zeigt an, welche Programme der Applikation auf Datenfiles zugreifen. Die grün dargestellten Knoten repräsentieren Datenfiles. Die Zahlenangaben an den Kanten geben die Anzahl gleichartiger Zugriffe an. Unterschiedliche Kantenarten repräsentieren lesende und schreibende Zugriffe.





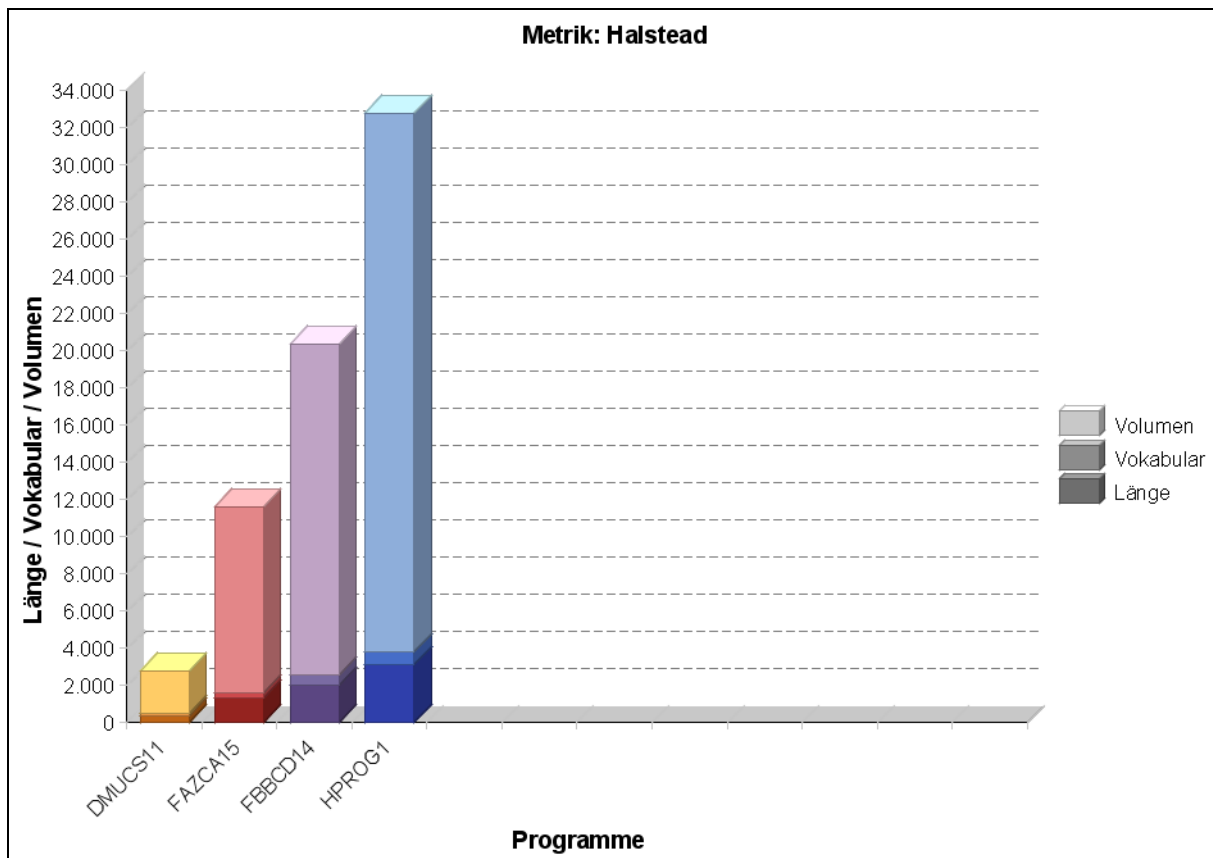
2.4 Metriken

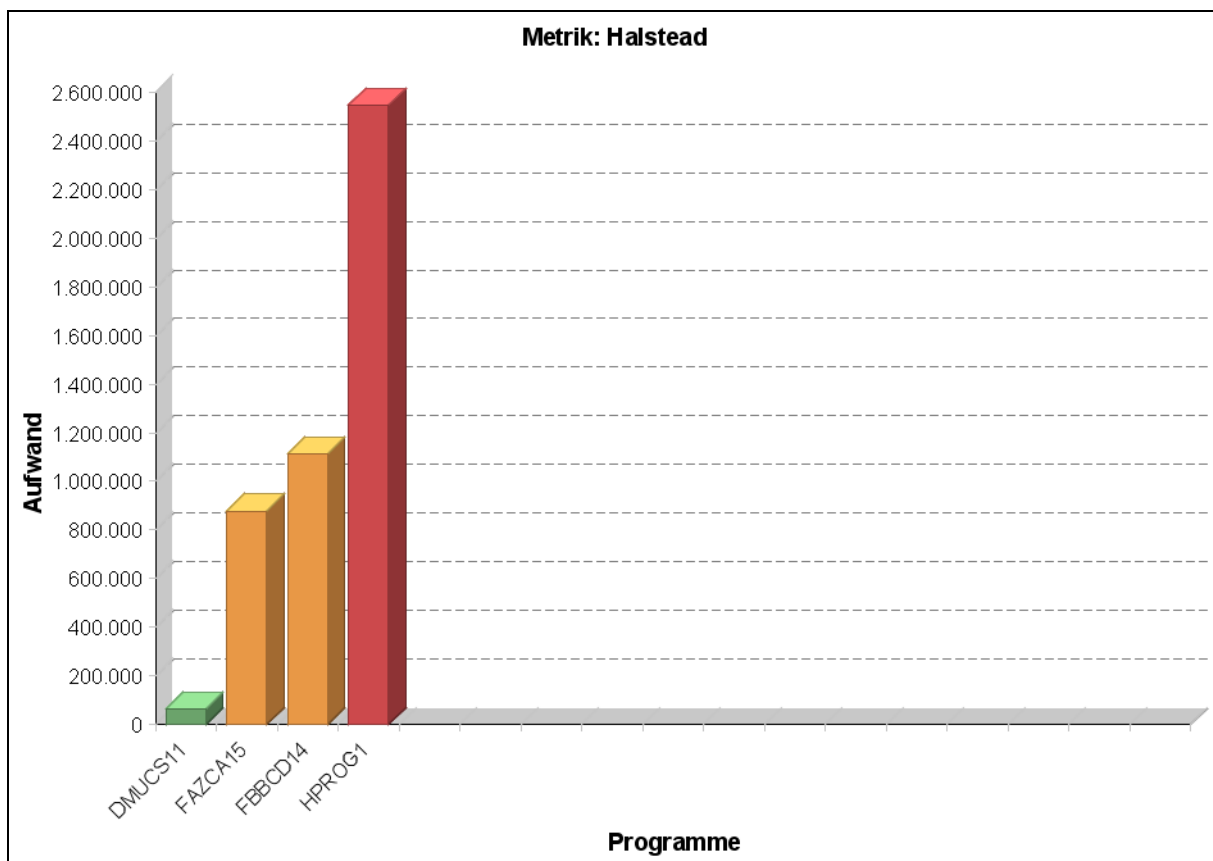
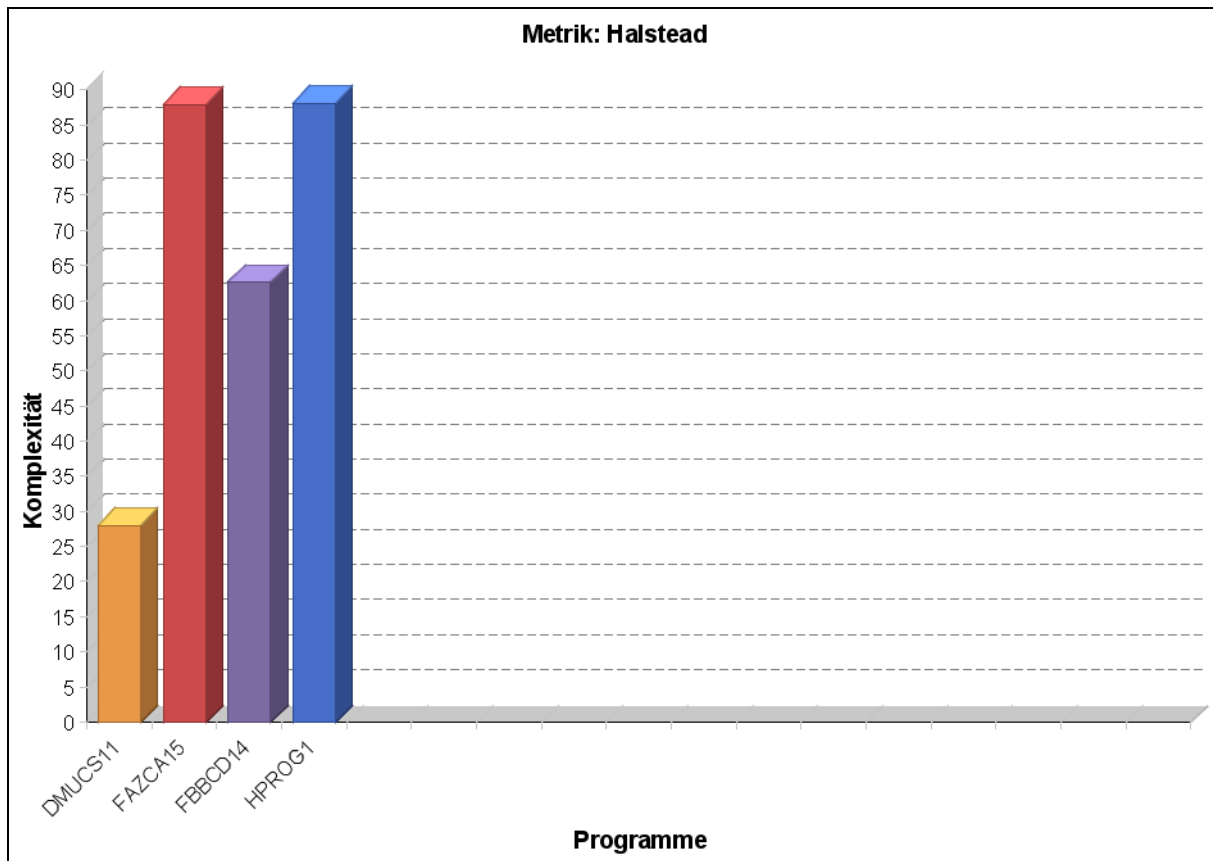
2.4.1 Metrik nach Halstead

Die Halstead-Metrik dient zur Messung der textuellen Komplexität und beschreibt den zu erwartenden Test- und Wartungsaufwand. Dabei werden folgende Werte ermittelt:

- Programmlänge: Summe der Operatoren und Operanden
- Programmvokabular: Summe der unterschiedlichen Operatoren und Operanden
- Programmvolumen: Größe der Programmimplementierung, basiert auf der Anzahl der durchgeführten Operationen und der genutzten Operanden.
- Programmschwierigkeit/Komplexität: Schwierigkeit, das Programm zu verstehen oder zu implementieren
- Programmieraufwand: Aufwand, das Programm zu implementieren

Die Grafik zeigt die Ergebnisse für die analysierten COBOL-Hauptprogramme:



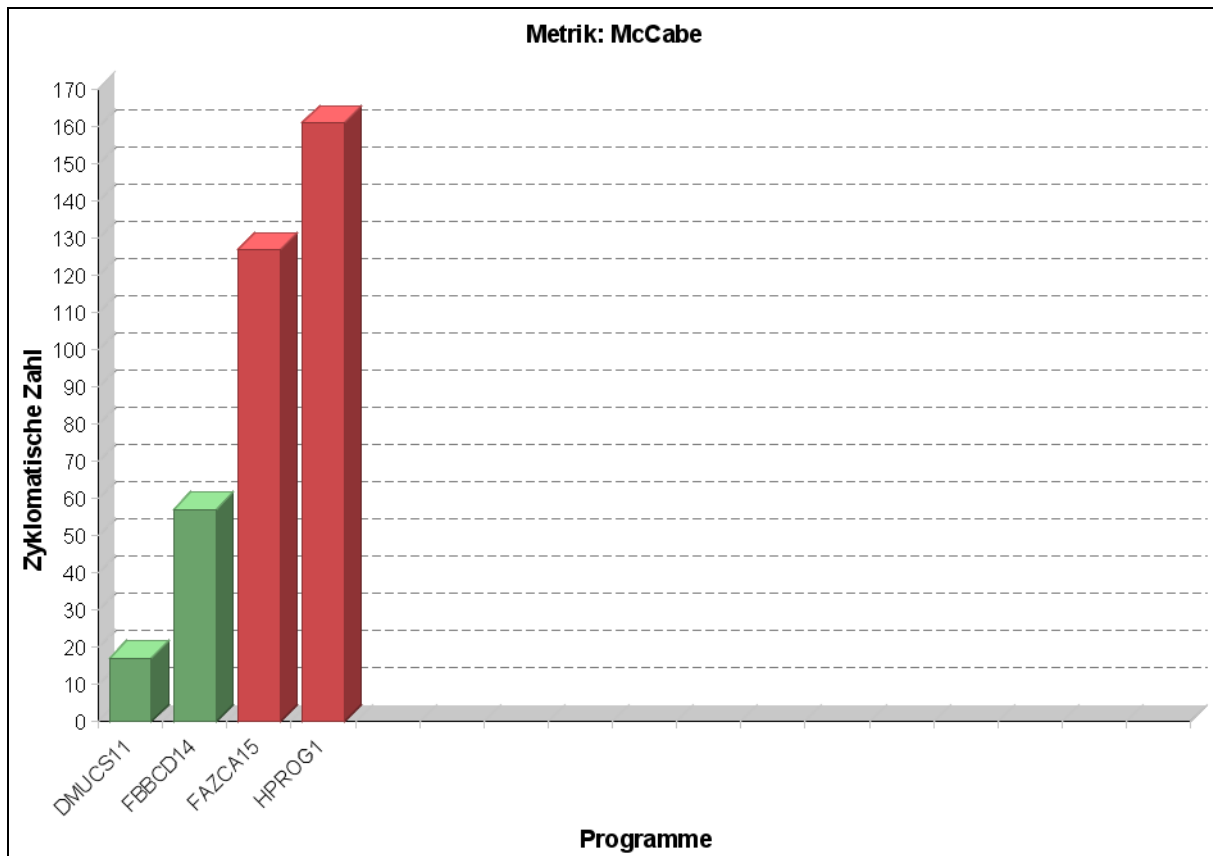




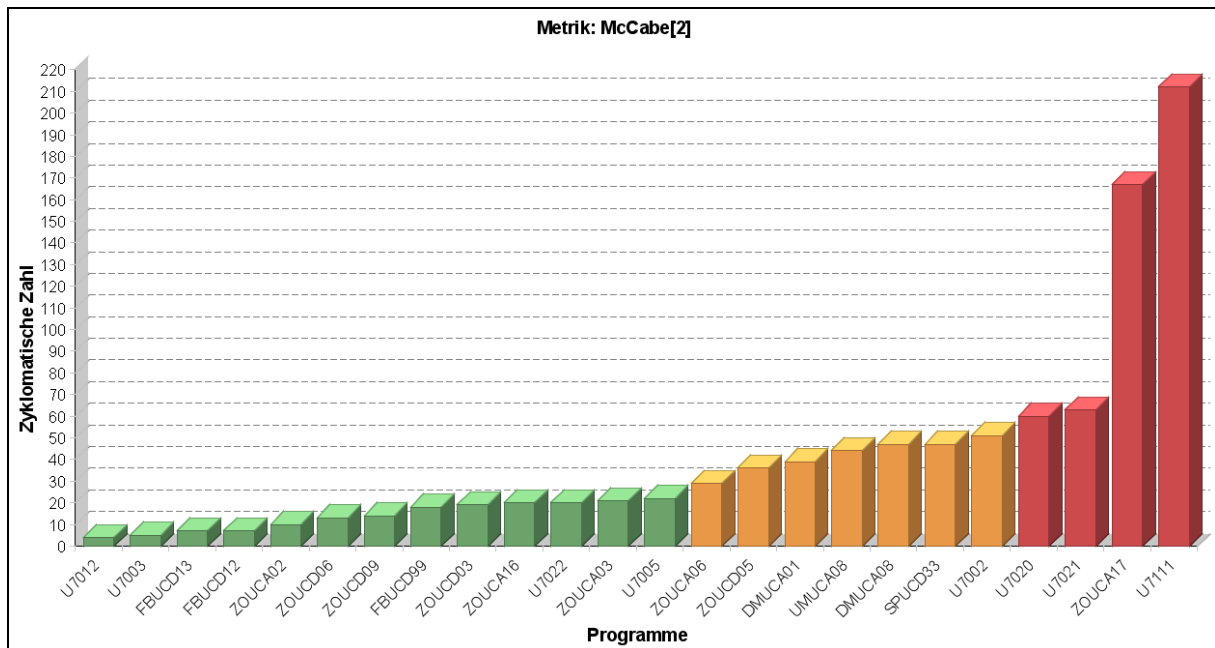
2.4.2 Metrik nach McCabe

Die zyklomatische Zahl nach McCabe ist ein Maß für die strukturelle Komplexität eines Programms. Dies beeinflusst die Testbarkeit, Wartbarkeit und Verständlichkeit eines Programms. Bei Werten > 10 steigt die Fehlerhäufigkeit überproportional.

Die Grafik zeigt die analysierten COBOL-Hauptprogramme. Die rot dargestellten Programme weisen eine zu hohe Komplexität auf.

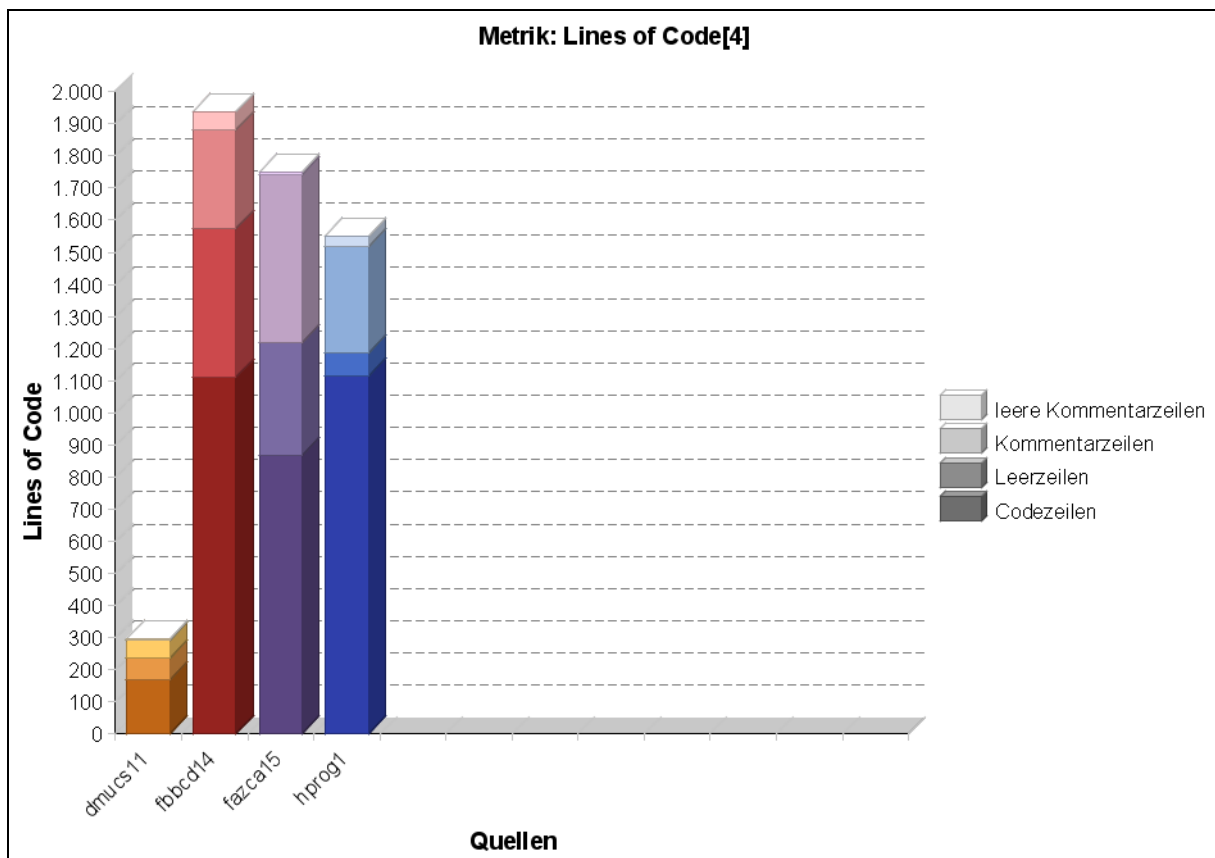


Die Grafik zeigt die analysierten Unterprogramme in der Bewertungsansicht:



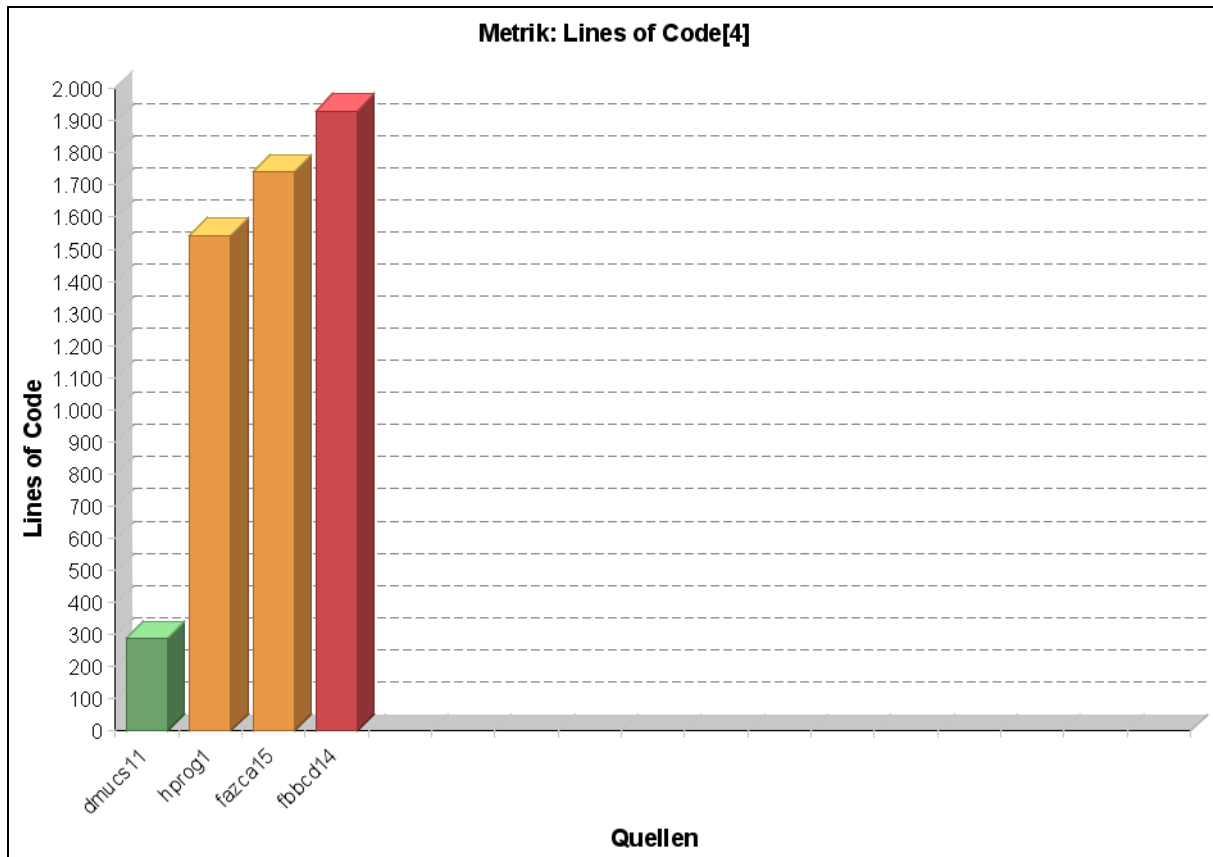
2.4.3 Metrik Lines of Code

Diese Metrik befasst sich mit der Ermittlung der Anzahl der Programmzeilen. Die Grafik zeigt die analysierten COBOL-Hauptprogramme in Detailansicht:





In der nachfolgenden Grafik sind die Hauptprogramme der Applikation in Bewertungsansicht dargestellt. Ein rot markierter Balken zeigt an, dass der Anteil der Kommentierung des Programms zu gering ist:





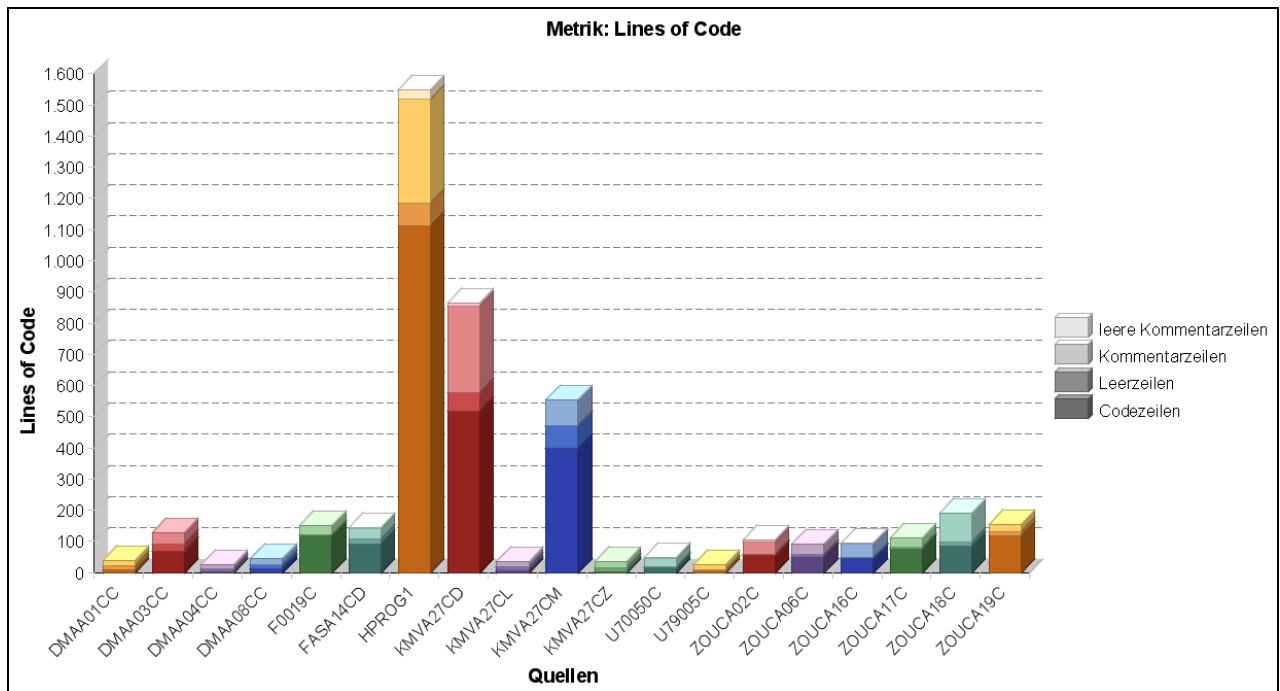
3 Feingranulare Analyse des Programm HPROG1

3.1 Programmstatistik

Quelle	Gesamtzeilen	Codezeilen	Leerzeilen	Kommentarzeilen	Leere Kommentarzeilen
Hauptprogramm					
HPROG1	1562	1108	71	354	29
Copybooks					
DMAA01CC	35	7	12	16	0
DMAA03CC	126	67	23	36	0
DMAA04CC	23	2	8	13	0
DMAA08CC	42	13	11	18	0
F0019C	149	120	0	29	0
FASA14CD	139	92	16	30	1
KMVA27CD	859	518	58	275	8
KMVA27CL	31	4	11	16	0
KMVA27CM	549	397	71	81	0
KMVA27CZ	34	3	12	19	0
U70050C	43	16	2	24	1
U79005C	20	5	1	14	0
ZOUCA02C	100	56	1	38	5
ZOUCA06C	90	49	9	32	0
ZOUCA16C	89	45	2	41	1
ZOUCA17C	107	73	5	29	0
ZOUCA18C	188	84	13	91	0
ZOUCA19C	153	117	13	23	0
Summe	4339	2776	339	1179	45



Die nachfolgende Grafik zeigt die Detailansicht des Mengengerüsts des Programms HPROG1 und aller zugehörigen Copybooks:



3.2 Metriken

- Metrik nach HALSTEAD:
 - Programmlänge: 3097 (Summe der Operatoren und Operanden)
 - Programmvokabular: 634 (Summe der unterschiedlichen Operatoren u. Operanden)
 - Programmvolumen: 28.828
 - Komplexität: 88
 - Programmieraufwand: 2.554.230
- Metrik nach McCabe:
 - Zyklomatische Zahl: 161
 - Bewertung des analysierten Programms: zu komplex

3.3 Genutzte Datenfiles

Datenfile-Name	Zugriffsart	Anzahl
PROTOKOLL	schreibend	20
LISTECPG	schreibend	20
LISTEDISK	schreibend	20

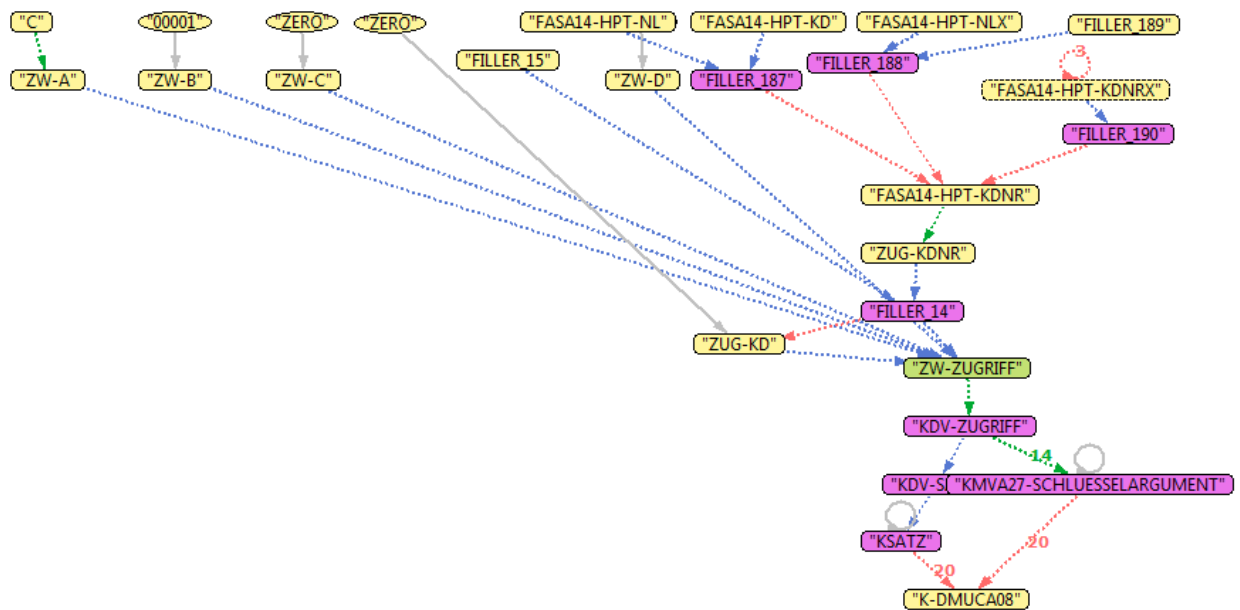


3.4 Genutzte Datenbanktabellen

Es werden keine Datenbanken genutzt.

3.5 Datenfluss eines ausgewählten Objekts

COBOL-Struktur: ZW-Zugriff

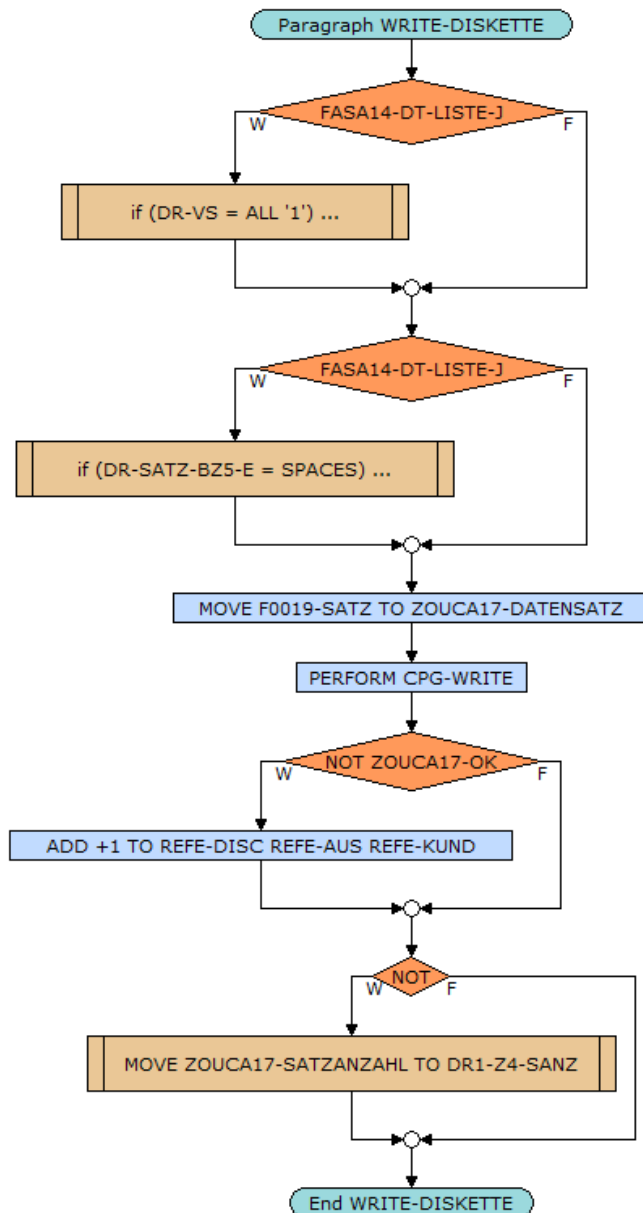




3.6 Steuerfluss eines ausgewählten Paragraphen

Paragraph: **WRITE-DISKETTE**

Anmerkung: Im dargestellten Steuerflussgraphen wurden einige Knoten zusammengeklappt, um eine bessere Übersichtlichkeit zu erreichen.





3.7 Aufrufhierarchie

Die rot gekennzeichneten Programmsymbole weisen auf ein gerufenes, jedoch nicht analysiertes Unterprogramm hin.

Die Zahlenangaben an den einzelnen Kanten geben die Anzahl der Aufrufe eines Unterprogramms durch ein Hauptprogramm an.

