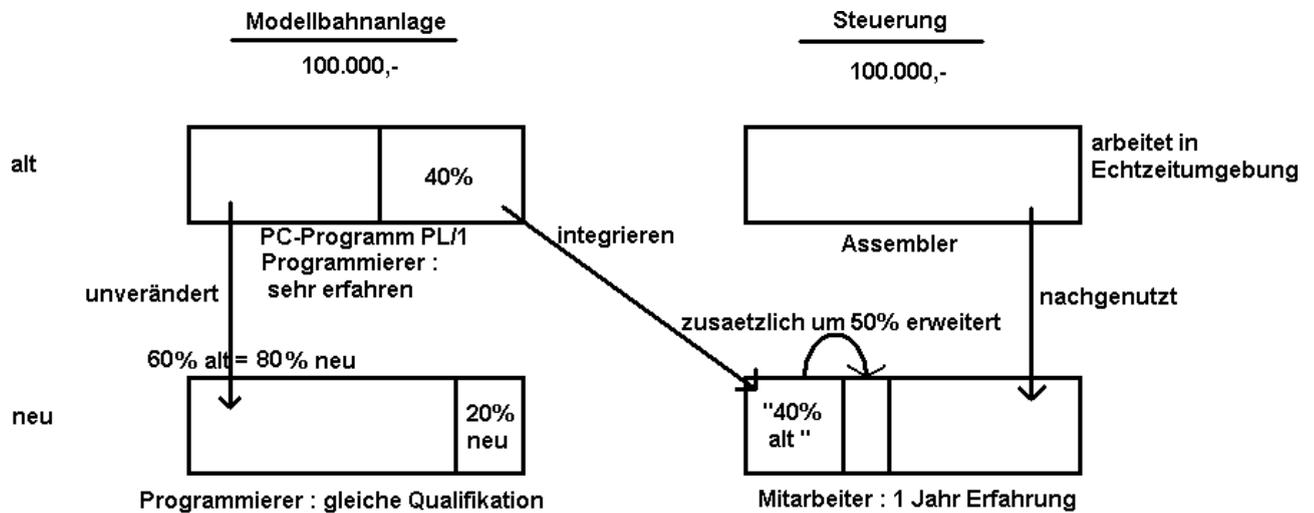


1.) Analogie- / Relationsmethode



1.a)

Kosten für neues Projekt :

PC-Programm PL/1 :

60% alt = 80% neu (1:1 übernommen) 0,-  
 20% neu 20.000,-

(eigentlich noch Nachkontrolle / Anpassung)

Assembler-Programm (Mikrorechner) :

Nachnutzung alter Software :  $\frac{1}{4} \cdot 100.000,- = 25.000,-$

„40% alt“ → neu

	neu (Assembler)	alt (PL/1)	Mehraufwand / Einsparung
Programmiersprache	140	100	140-100 = 40
Erfahrung Programmierer	1 Jahr = 140	sehr erfahren = 5 Jahre = 80	140-80 = 60

⇒ Mehraufwandsfaktor = 40+60 = 100(%) → Relationsmethode

d.h. „40% alt“ = 40.000,-  
 + Mehraufwand = 40.000,-  
 = 80.000,-

dazu von diesem 50% Erweiterung , d.h. 50% von 80.000,- = 40.000,-

⇒ Kosten fuer die Steuerung :

25.000,-  
 + 80.000,-  
 + 40.000,-  
 = **145.000,-**

### 1.b)

$$\text{Gesamtsumme} = 20.000,- + 145.000,- = 165.000,-$$

in Echtzeit bereits vorhanden : 25.000,- (Nachnutzung)  
neu in Echtzeit bereitstellen : 120.000,- , d.h.  $120.000 \cdot 1,8 = 216.000$   
 $\Rightarrow 241.000,-$

$$\Rightarrow \text{Mehrkosten} : 241.000 - 165.000 = 76.000,-$$

### 1.c)

3 Mikrorechner

Software einheitlich , bis auf die „50%“ Neuentwicklung  
1x bereits vorhanden

$$\Rightarrow \text{Mehrkosten} : 2 \cdot \frac{1}{4} \cdot 40.000 = 20.000,-$$

### 2.)

Systemanalytiker = d  
Programmierer = a  
Qualitätssicherer = c  
Software-Ergonom = b

### 3.) Zielanalyse

- Hierarchiebaum aufstellen (Teilziele , Oberziel)
- Wichtung der Teilziele :
  - relative Wichtigkeit der Teilziele zum Oberziel vorgeben
  - relative Wichtigkeit der Teilziele zum Gesamtziel vorgeben

$$\text{Knotengewicht}[\%] = \text{Stufengewicht} \cdot \frac{\text{Knotengewicht des Oberziels}}{100}$$

$$\sum_{\text{K ist Kind vom Vater X}} \text{Knotengewicht von K} = \sum_{\text{K}} \text{Stufengewicht von K} \cdot \frac{\text{Knotengewicht des Oberziels}}{100}$$

$$= \text{Knotengewicht des Oberziels} \cdot \sum_{\text{K}} \frac{\text{Stufengewicht von K}}{100}$$

