

Wyznaczanie złożoności komponentów

FTR's	DATA ELEMENTS		
	1-5	6-19	> 19
0-1	Low	Low	Ave
2-3	Low	Ave	High
> 3	Ave	High	High

FTR's	DATA ELEMENTS		
	1-4	5-15	> 15
0-1	Low	Low	Ave
2	Low	Ave	High
3 or more	Ave	High	High

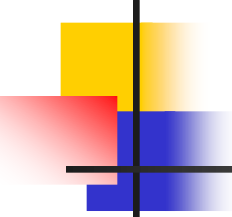
RET's	DATA ELEMENTS		
	1-19	20 - 50	> 50
1	Low	Low	Ave
2-5	Low	Ave	High
> 5	Ave	High	High

Rating	VALUES		
	EO	EQ	EI
Low	4	3	3
Average	5	4	4
High	7	6	6

Rating	Values	
	ILF	EIF
Low	7	5
Average	10	7
High	15	10

Złożoność komponentów -> UFP

Type of Component	Complexity of Components			
	Low	Average	High	Total
External Inputs	x 3 =	x 4 =	x 6 =	
External Outputs	x 4 =	x 5 =	x 7 =	
External Inquiries	x 3 =	x 4 =	x 6 =	
Internal Logical Files	x 7 =	x 10 =	x 15 =	
External Interface Files	x 5 =	x 7 =	x 10 =	
Total Number of Unadjusted Function Points				
Multiplied Value Adjustment Factor				
Total Adjusted Function Points				



VAF (Value Adjustment Factor)

$$\text{VAF} = 0,65 * [\Sigma F_i / 100]$$

F1 – wymiana danych

F2 – rozproszenie przetwarzania

F3 – wydajność aplikacji

F4 – wymagania sprzętowe

F5 – ilość transakcji, duże obciążenie aplikacji

F6 – wprowadzanie danych online w trakcie działania aplikacji

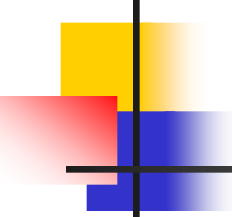
F7 – efektywność wprowadzania danych przez użytkowników

F8 – modyfikacje danych online (np. bazy danych)

F9 – złożoność przetwarzania danych

F10 – możliwość wykorzystania kodu aplikacji (modularność)

F11 – łatwość instalacji



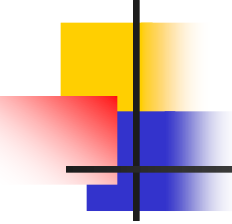
VAF (Value Adjustment Factor)

F12 – łatwość administracji

F13 – instalacja w wielu lokalizacjach

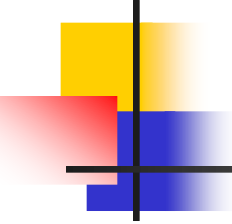
F14 – łatwość pielęgnacji i dostosowania

$$\mathbf{FC = UFC * VAF}$$



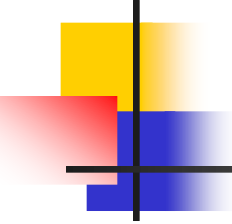
Średnia wartość LOC/FP

Język	Poziom języka	Średnia wartość LOC/FP
Asembler	1	320
C++	6	53
Delphi	11	29
Turbo Pascal > 5	6,5	49
Visual C++	9,5	34
Java	6	53
Power Builder	20	16



Średnia Pracochłonność/FP

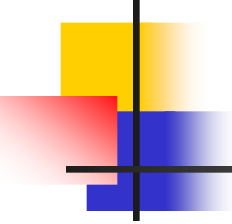
Poziom języka	Średnia ilość FP na jeden osobomiesiąc
1-3	5-10
4-8	10-20
9-15	16-23
16-23	15-30
24-55	30-50
Powyżej 55	40-100



Przykład

Aplikacja do generowania i drukowania faktur VAT na podstawie CDR'ów. Aplikacja powinna posiadać następującą funkcjonalność:

- Import plików z połączeniami CDR
- Wprowadzenie danych o kontrahencie
- Wprowadzenie danych o numerach telefonicznych kontrahenta
- Generowanie faktur VAT za połączenia
- Drukowanie faktur VAT
- Drukowanie listy obecnych kontrahentów
- Drukowanie listy numerów telefonicznych



Przykład

W wyniku analizy określono liczbę punktów funkcyjnych:

$$\mathbf{FC = 58}$$

Zakładając, że aplikacja zaimplementowana zostanie w języku Java, wyznaczono liczbę linii kodu:

$$\mathbf{LOC = 58 * 53 = 3074}$$

Dla języka o kategorii 4-8, jedna osoba w ciągu miesiąca może zaimplementować od 10-20 FP, co oznacza:

- minimalny czas realizacji aplikacji $58/20 = 2,8$
- maksymalny czas realizacji aplikacji $58/10 = 5,8$