Chapter - 4

* S/W size $\xrightarrow[\text{by}]{\text{measured}}$ LOC
  ↓
  number of lines of code

* SEI – Software Engineering Institute
* NKLOC – non-commented thousand LOC
* LLOC – logical lines of code

rules

What is Language Gearing Factor?

→ compares the expressiveness of languages and takes into account the differences in productivity in languages.

more productive → fewer LOC

* Initially, gearing factor tables showed relationship between LOC and function points

* Many gearing factor value: (1991/2005)

C – 128/148 Java – 53/60

C++ – 55/60 Assembly-basic – 320

streamstech

* Reused code classification scheme has an ordinal scale with 4 points

1) Reused verbatim
2) Slightly modified (<25%)
3) Extensively modified (≥ 25%)
4) New

Simplified : 1) reused (<25% changed)
2) new

Issue of LOC :
→ only tells size of code
→ not tell if the problem is solved.

Solution: Function Points (FP)
→ count the I/P, O/P, interfaces and DB

FPA - Funtion Point Analysis
IFPUG - International Function Point Users Group

* UFP – unadjusted function points
  – ~~sum~~ counted by summing all of the complexity ratings.

* VAF – value adjustment factor
  – calculated based on the complexity of the overall system.

* AFP – adjusted function points
  ~~counted by mul~~

$$\boxed{AFP = UFP \times AFP}$$

* FP complexity ratins:

5 components:
1) Inputs (I)
2) Outputs (O)
3) Data Files (F)
4) Interfaces (N)
5) Inquiries (Q)

3 types: simple, average, complex

* FP complexity Rating value :

	simple	Avg	complex
I	3	4	5
O	4	5	7
F	7	10	15
N	5	7	10
Q	3	4	6

www

GUI – Graphical User Interface

Ex: You've small program
4 simple inputs
1 data file (trivial)
2 outputs – average

How many unadjusted function points would this be? UFP

→ Inputs = 4 x 3 = 12

Outputs = 2 x 5 = 10
Data file = 1 x 7 = 7
∴ Total UFP = 12 + 10 + 7 = 29

VAF is calculated based on GSCs (total 14)
GSC → General Systems Characteristics
rate: 0 to 5 scale
0 → not present / no influence
5 → more influence

IFPUG standard

AFP = UFP x (0.65 + 0.01 x VAF)

streamstech

GSC:

1) Data Communications
2) Distributed data /processing
3) Performance Objectives
4) Heavily Used configuration
5) Transaction Rate
6) Online Data Entry
7) End-user efficiency
8) Online update
9) Complex processing
10) Reusability
11) Conversion / Installation Ease
12) Operational Ease
13) Multiple Site Ease
14) Facilitate Change

www.streamstech.com

streamstech

Ex: small Project UFP – 25

case : n. all trivial (simple) – 1

total = 14

2 { complex : processing
data communication } 5

How many AFP?

→ UFP = 25

VAF = (12 × 1 + 2 × 5)

= 12 + 10 = 22

∴ AFP = UFP × (0.65 + 0.01 × VAF)

= 25 × (0.65 + 0.01 × 22)

≈ 21.75

SPR (software Productivity Research)

↳ VAF calculation:

2 factors:

1) Problem complexity (PC)

2) Code/ complexity (DC)
Data

Ex: UFP = 35

using SPR FP complexity adjustment method

→ many difficult algo (5)

→ simple data with few elements and relations. (1)

→ what is the $\boxed{AFP}$?

www.streamstech.com

→ $AFP = UFP \times (0.4 + 0.1(PC + DC))$

$= 35 \times (0.4 + 0.1(5+1))$

$= 35$

* Program Complexity

1) algorithm and calculations
2) Complexity

Data Complexity

1) Data (elements and relationships)
2) file, field, variable

Pros and Cons of FP:

Pros

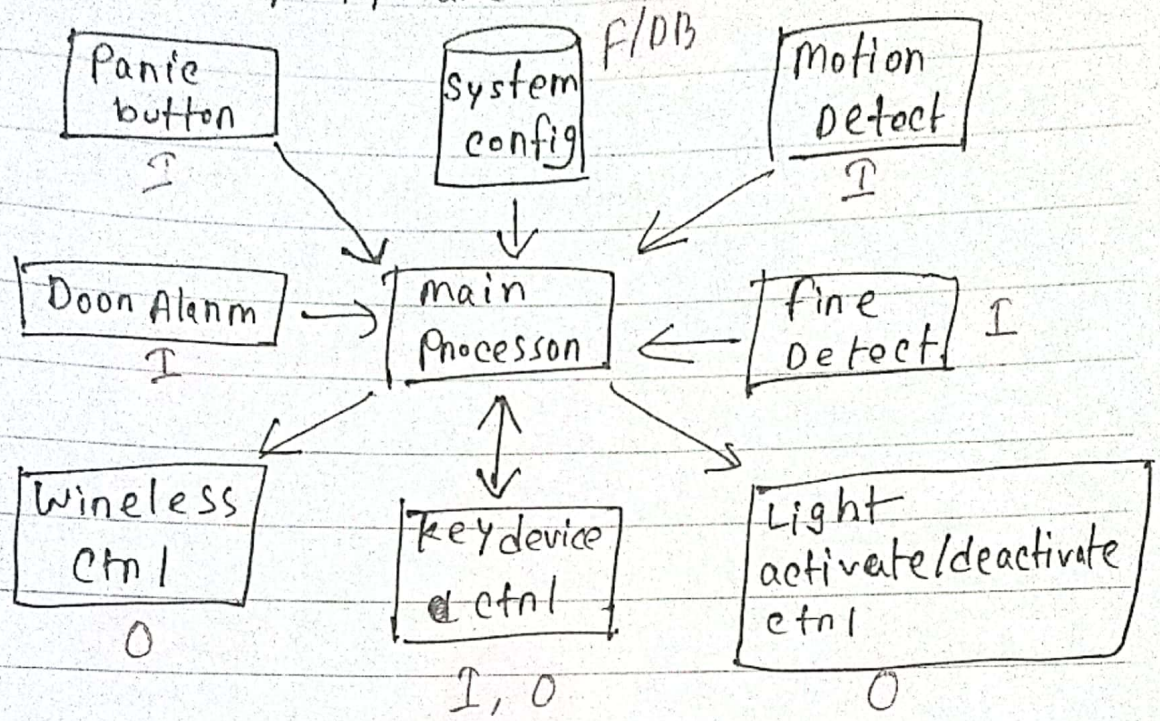
- technology indepent
- effective early phase (requirement)
- well-documented, specified
- follow standards
- a methodology
- reliable, accurate
- useful in user mgmt.

Cons

- difficult to understand
- many steps
- no tools to automatically count FP
- more effort needed

4.2.1.2 Ex:

* Home safety archi



→ Input = 5 Simple

Output = 2 Simple Database = 1 Simple

Medium Output = 1

∴ UFP = 5 × 3 + 2 × 4 + 1 × 5 + 1 × 7

= 15 + 8 + 5 + 7

= ~~38~~ 35

GSC (assume)

Data communications	3	Transaction	0
Distributed	2	Online Data Entry	1
Performance	3	End-user efficiency	1
Heavy Config	1	Online update	1
Installation ease	5	Complex Processing	1
		Reusability	3

streamstech

operational ease - 5 Facilitation of change
multiple sites - 5 - 2

$\therefore$ VAF = 33

$\therefore$ AFP = 35 × (0.65 + 0.01 × 33)

= 34.3

(a) Estimate the effort to build the system?

10 FP / staff month

SD : ± 1 staff month

→ estimated number of staff month = 34.3/10 = 3.43

$$SD = \frac{34.3}{10-1} = 3.81$$

$$\text{or } \frac{34.3}{10+1} = 3.12$$

So, effort is between ~3.12 and ~3.8 staff month

www.streamstech.com

b) Now you're told that company can only affond 75% of the estimated staff months to build system. What do you do now?

20% unpaid overtime → not get much credit

↳ reduce AFP
Change FP > impact on estimate

→ Reduce expected staff month 25% (min)

→ reduce AFP

Step 1: UFP simplify

↳ not reduce functionality

↳ reduce inputs

4 × 3 = 12 ← instead of 4 separate inputs (simple)

1 × 4 = 4 ← → 1 average input

Sol Before, AFP = 34.3

Now, ~~AFP~~ UFP = 4 + 3 + 8 + 5 + 7
= 27

∴ AFP = 27 × (0.65 + 0.01 × 33)
= 26.46

streamstech

<u>Step 2:</u> Faster processor,

Before, GSC = 3

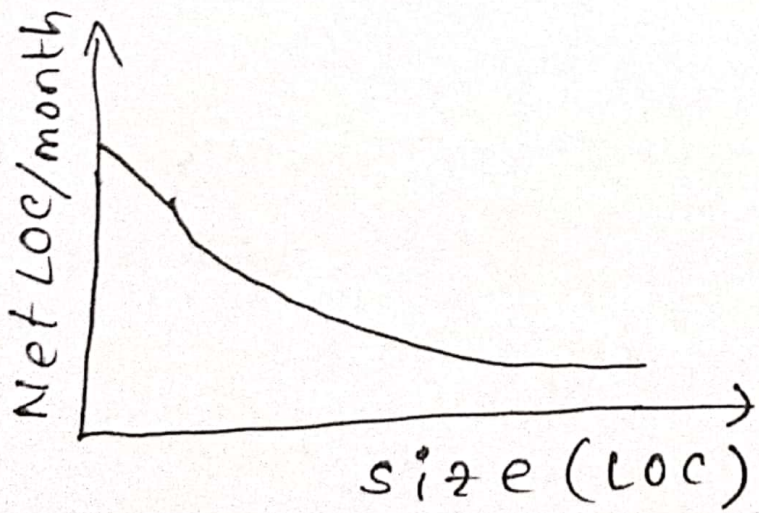
Now, GSC = 1

$\therefore$ VAF = 31

$\therefore$ AFP = $27 \times (0.65 + 0.01 \times 31)$

$= 25.92$

*



* Feature Point's weights:

Algo - 3

I - 4

O - 5

Q - 4

F - 7

N - 7

www.streamstech.com