

Function Points, Backfiring, and Costar

Dan Ligett
Softstar Systems

Ligett@SoftstarSystems.com
<http://www.SoftstarSystems.com>
(603) 672-0987

Copyright (c) 2003 Softstar Systems

1

Costar Approaches to Backfiring

- ☆ Costar has let you do backfiring since 1986
- ☆ Use predefined language & backfiring ratio
- ☆ Define your own language & backfiring ratio
- ☆ Supply a backfiring ratio (i.e. ignore language)

Copyright (c) 2003 Softstar Systems

2

Costar - Sample (Component 1)

File View Reports Components Tools Preferences Help

Estimate: Sample ID: Model: COCOMO II 2000

Component: Component 1 ID: Increment: 1

Totals for entire Project	Effort (PM)	Duration (Mo)	Cost (K\$)	Productivity	Equivalent Size
Requirements	RQ	0.7	1.2	6.2	
Development	PD+DD+CT+IT	9.8	7.6	304.8	Total Size
Total	RQ+PD+DD+CT+IT	10.5	8.8	94.8	3,000

	Simple	Average	Complex
External Input	10	0	0
External Output	0	0	0
Logical Internal File	0	0	0
External Interface File	0	0	0
External Inquiry	0	0	0
UnAdjusted Function Points	30		
Adjusted Function Points	30		
Lines per Function Point	100		
Size before REVL	3,000		

Buttons: Change FP Adjustment Factor... Change Lines per Function Point...

Drivers & Size / Model / REVL / Reuse / Function Points / Increments / Breakage / Costs / Rates / Maint / Filter / Descr

Enter the number of Single External Inputs

Sample: 10.5 PM, 8.8 Months Component: 10.5 PM EAF: 1.0000 Level: 1

Costar – screen shot

Lines per Function Point Worksheet

Source of Lines per Function Point

- Inherit
- Enter Lines per Function Point: 100
- Select Language from List

Development Language

OK Cancel Help

Calico – screen shot

Calico

File Equation Cost Driver Distribution Function Point Customize Help

COCOMO Model Name: COCOMO_II_2000_CUSTOM Model ID: 2001

SLOC per Function Point

Enter Default Lines of Code per Function Point:

Language	Lines of Code per Function Point	Language	Lines of Code per Function Point
C	125	Ada	70
Fortran	100	C++	53
COBOL	100	Java	51
Basic	95	Visual C++	14
Pascal	85	LatIn	5
PL/I	75		

Copyright (c) 2003 Sofistar Systems

5

Calico – screen shot

Calico

File Equation Cost Driver Distribution Function Point Customize Help

COCOMO Model Name: COCOMO_II_2000_CUSTOM Model ID: 2001

Function Point Weight

	Simple	Average	Complex
External Input	3	4	6
External Output	4	5	7
Logical Internal File	7	10	15
External Interface File	5	7	10
External Inquiry	3	4	6

Copyright (c) 2003 Sofistar Systems

6

Backfiring Choices

- ☆ Use silly old ratios
- ☆ Use silly new ratios
- ☆ Calculate your own ratios

Costar Backfiring Summary

- ☆ Do whatever works
- ☆ Calibrate your own
- ☆ Count your own – it's easy
- ☆ Costar lets you enter new ratio
- ☆ Calico lets you build in new languages
- ☆ Call us – unlimited technical support

Advice

Do it yourself!

- ☆ The published backfire ratios are suspect
 - We don't have access to the original data
 - They didn't use COCOMO II counting rules
 - Very precise, maybe not so accurate
 - Some of the numbers are very old
- ☆ It's easy to find a SLOC counter
- ☆ It's relatively easy to count FP

Ignore the published ratios -- calculate a ratio for
Your coding practices, your toolset, your domain.