

The Origins of Function Point Metrics

Capers Jones

05/17/2016

VP and CTO, Namcook Analytics LLC

Email: Capers.Jones3@gmail.com

Introduction

The author was working at IBM in the 1960's and 1970's and was able to observe the origins of several IBM technologies such as inspections, parametric estimation tools, and function point metrics. This short paper discusses the origins and evolution of function point metrics.

In the 1960's and 1970's IBM was developing new programming languages such as APL, PL/I, PL/S etc.

IBM executives wanted to attract customers to these new languages by showing clients higher productivity rates.

As it happens the compilers for various languages were identical in scope and had the same features but some older compilers were coded in assembly language while newer compilers were coded in PL/S, which was a new IBM language for systems software.

When we measured productivity of assembly-language compilers versus PL/S compilers using "lines of code" we found that even though PL/S took less effort, LOC per month favored assembly language.

To show the value of higher-level languages the first IBM approach was to convert high-level languages into "equivalent assembly language." In other words we measured productivity against a synthetic size based on assembly language instead of against true LOC size in the higher level languages. This method was used by IBM from around 1968 through 1972.

An IBM vice president, Ted Climis, said that IBM was investing a lot of money into new and better programming languages. Neither he nor clients could understand why we had to use the old assembly language as the metric to show productivity gains for new languages. This was counter-productive to the IBM strategy of moving customers to better languages. He wanted a better metric that was language independent and could be used to show the value of all IBM high-level languages.

This led to the IBM investment in function point metrics and the creation of a function-point development team under Al Albrecht at IBM White Plains.

Function Point metrics were developed by the IBM team by around 1975 and used internally and successfully. In 1978 IBM placed function point metrics in the public domain and announced them via a technical paper by Al Albrecht at a joint IBM/SHARE/Guide conference in Monterey, California.

Table 1 shows the basis of the IBM function point invention based on the early comparison of assembly language and PL/S for IBM compilers.

Table 1 shows productivity in four separate flavors: 1) actual LOC in the true language; 2) productivity based on “equivalent assembly code”; 3) productivity based on “function points per month”; 4) productivity based on “work hours per function point.”

Table 1: Function Point Evolution Circa 1968-1975

(Results for two IBM compilers)

	Assembly Language	PL/S Language
Lines of code (LOC)	17,500.00	5,000.00
Months of effort	30.00	12.50
LOC per month	583.33	400.00
Equivalent assembly language	17,500.00	17,500.00
Equiv. Assembly per month	583.33	1,400.00
Function point sizes	100.00	100.00
Function Points per month	3.33	8.00
Work hours per FP	39.60	16.50

The creation and evolution of function points was based on a need to show IBM clients the value of IBM’s emerging family of high-level programming languages such as PL/I, APL, and others. This is still a valuable use of function points since there are more than 2,500 programming languages in 2016 and new languages are being created at a rate of more than one per month.

Once function points were released by IBM in 1978 other companies began to use them, and soon the International Function Point User’s Group (IFPUG) was formed in Canada.

Today in 2016 there are hundreds of thousands of function point users and hundreds of thousands of benchmarks based on function points. There are also several other varieties of function points such as COSMIC, FISMA, NESMA, etc.

Overall function points have proven to be a successful metric and are now widely used for productivity studies, quality studies, and economic analysis of software trends. Function point metrics are supported by parametric estimation tools and also by benchmark studies. There are also several flavors of automatic function point tools. There are also function point associations in most industrialized countries.

Table 2 shows countries with increasing function point usage circa 2016, and it also shows the countries where function point metrics are now required for government software projects.

Table 2: Countries Expanding Use of Function Points 2016

1	Argentina	
2	Belgium	
3	Brazil	Required for government contracts 2008
4	Canada	
5	Finland	
6	France	
7	Germany	
8	India	
9	Italy	Required for government contracts
10	Japan	Required for government contracts
11	Malaysia	Required for government contracts
12	Mexico	
13	Peru	
14	Singapore	
15	South Korea	Required for government contracts
16	Spain	
17	Switzerland	
18	The Netherlands	
19	United Kingdom	
20	United States	

In retrospect function point metrics have proven to be a powerful tool for software economic and quality analysis.