

August 2017

INTERNATIONAL
FUNCTION POINT
USERS GROUP

191 Clarkville Road
Princeton Junction, NJ 08550
USA



Vol. 11 • Issue 2

Metric Views

www.ifpug.org

Celebrating IFPUG's

30th

Anniversary

Inside...

In This Edition.....	2
Letter from the President.....	2
From the Editor.....	3
IFPUG 30 years – International Year of Software Measurement Timeline	4
When Metrics Mean Business	6
An Interview with Capers Jones	10
Thirty Years of IFPUG.....	13
Simple Function Points.....	15
Remembering Alan Albrecht.....	17
Nesma Congratulates IFPUG on Their 30th Anniversary	21
Board of Directors.....	22
The COSMIC Functional Size Measurement Method.....	22
The International Software Benchmarking Standards Group Congratulates IFPUG	23
Committee Rosters.....	25
CFPS Certification Listing	26
CFPP Certification Listing	26
Certifications Matters.....	26
Fellows.....	27
CSP Certification Listing.....	27

IN THIS EDITION

We are excited to present you with this Anniversary edition of *Metric Views*. Join us in celebrating the 30th Anniversary of IFPUG and the International Year of Software Measurement (IYSM).

IFPUG takes pride in being at the forefront of functional measurement for the past 30 years. Our representation in the international community has grown significantly over the years. In celebration of our growing international community we have included insightful and informative articles from Brazil and Italy. Mauricio Aguiar discusses the business drivers that have made Brazil the number one function point user in the world and Roberto Meli advances the idea that the time is right for functional metrics to evolve towards simplification and agility.

To celebrate the International Year of Software Measurement, we have included articles that inform us of alternative functional measures that are being successfully used in addition to function points. Two alternative approaches to sizing, COSMIC and Nesma, are presented in this edition.

As a special treat we have included a recent interview with Capers Jones. Often referred to as a measurement 'guru', Capers is a well-known author and international public speaker. He has written numerous books on software measurement and function points.

Throughout this edition you will see various reflections from various IFPUG presidents. These men and women have given their time and talents to keep this vibrant users group relevant in today's software industry.

A picture is worth a thousand words and throughout this edition you will see photos from past IFPUG conferences. We hope you enjoy this special edition of *Metric Views*. For you 'old timers', it will be trip down memory lane as you reminisce through the timeline of IFPUGs history. Enjoy

Message from the President



Tom Cagley

In 1986, Bill Huffs Schmidt was elected the first president of IFPUG. Thirty years later, as I near the end of my second term as IFPUG President, I invite you to join us as we celebrate the 30th anniversary of the International Function Point Users Group. Function Points are an internationally standardized unit of measure used to represent software size. While this definition describes the basic idea which IFPUG is built upon, there would be no reason to celebrate if that was all. This year we celebrate thirty years of education, standards, and community in the realm of functional size and non-functional size measurement. The thirty years since the founding of IFPUG have been marked by invaluable growth and change.

The last two years serving as President have afforded me the opportunity to meet IFPUG members and supporters in Italy, Poland, and, most recently, India. I still believe that IFPUG has the most value when members are able to network with other members. This was no more evident than during our trip to Mumbai,

India for the ISMA¹³ conference this past March. The IFPUG Board of Directors and I received the warmest welcome from the Computer Society of India and it was clear that our Indian members were pleased we had traveled a great distance to be a part of their workshops and conference. Building these strong relationships are what keeps IFPUG a vibrant and relevant presence in the measurement community.

In 2017, we celebrate years of continued education through workshops and conferences, the evolution of the Software Non-Functional Assessment Process (SNAP) sizing standard, the advent of online CFPS and CSP exams with a new testing partner, and organizational growth and awareness brought about through interorganizational involvement and promotion. I would ask that each of you, as members, continue to grow IFPUG as we move on to our next chapter. Complete a volunteer form, and then reach out to the chairs of the committee or committees you are interested in becoming involved with. If you are not interested in joining a committee but have an idea that will help IFPUG grow please reach out to us at ifpug@ifpug.org.

On behalf of the IFPUG Board of Directors and Committee Members, I thank you for celebrating IFPUG's 30th Anniversary as we continue on the path of successful measurement practices.

Sincerely,

Tom Cagley
IFPUG President



Paul Radford

From the Editor's Desk

Thirty years of IFPUG is celebrated in this edition. This is a celebration both of what has been achieved and where IFPUG is at now.

IFPUG is an enduring symbol for and champion of software measurement. IFPUG standards are the most widely recognized and applied across the world. These achievements are substantial and have been the result of the contribution and co-operation of hundreds of people over thirty years. Certainly, this is a time to celebrate the work, the milestones and the people who have made this possible.

But it is also a time for reflection.

Where have we fallen short? What have we failed to do, to attempt or to achieve?

From a personal point of view, many of my hopes for IFPUG have not been realised. In the beginning, I thought the obvious issues with Counting Practices would be addressed and

resolved in a short period. From that basis, IFPUG could then address technological issues (as many estimating tools of the period were already doing). Then a true integration into early processes would enable better, cheaper software development and consequent high level management awareness.

None of those things have been achieved. Many of the obvious issues with Counting Practices are still there. And we have added some new ones. SNAP may be a way forward – but it is not as I anticipated and I have some of the same basic issues with SNAP fundamentals as with some of the more bizarre aspects of IFPUG defined function point analysis rules and guidelines.

But our biggest problem has always been that function point sizing is seen as a producer of a magic number, rather than an integral and enormously useful view of software. Extending this understanding to the world of software business and development is an enormously difficult task.

IFPUG has been a constant in software measurement for 30 years – and everything is not yet perfect. However, the task has been, and still is, formidable.

IFPUG is still here. And maybe this process was always going to be a lengthy one. It is where we start from NOW that matters. And how we plan to mould the future.

And the fact is that IFPUG has achieved an enormous amount.

David Herron put it best:

“..... Thirty years ago we did not have COSMIC, we did not have NESMA, we did not have SIFPA, etc. All those alternative measurement practices evolved from IFPUG directly or indirectly. Even though we advertised as being an international users group we certainly did not have the international presence 30 years ago that we have today. This is what I think we should be celebrating. A celebration of software measurement evolution for the greater good, which includes a diversity of measurement approaches and diversity of populations putting measurement practices to use.”

This is the spirit and future intention that has been forged over 30 years of steps and mis-steps. IFPUG most certainly has not always held these views. Maturity brings perspective and IFPUG, hopefully, has arrived at a significant milestone. Whether it can now forge a real plan and co-operation with others still lies in the future. And, as new centres of measurement inspiration arise – as they always do - new alliances and ways forward may be required. This has been a long and difficult path and much of it is still in front of us.

We need to start with a few big steps.

Paul Radford
Communications and Marketing Committee

MetricViews

Published twice a year by the International Function Point Users Group (IFPUG), headquartered in Princeton Junction, New Jersey, U.S.A.

MetricViews August 2017

Editors

David Herron
Paul Radford

IFPUG Board of Directors

President

Tom Cagley

Vice President

Mauricio Aguiar

Secretary and Director of Certifications

Christine Green

Treasurer

Debra Maschino

Immediate Past President

Kriste Lawrence

Director of

Communications & Marketing

Carol Dekkers

Director of Sizing Standards

Dácil Castelo

Director of International Membership Committee

Pierre Almén

Director of Education & Conference Services

Luigi Buglione

IFPUG Office

Executive Director

Megan Capie

Association Coordinator

Michele Giovine

Membership Coordinator

Nicole Lauzon

Views and opinions presented in *MetricViews* articles may not represent those of the International Function Point Users Group (IFPUG).

Please submit all articles, news releases and advertising to:



IFPUG/MetricViews

191 Clarksville Road

Princeton Junction, NJ 08550

(609) 799-4900

ifpug@ifpug.org

IFPUG 30 years – International Year of Software Measurement Timeline...

Compiled by Carol Dekkers, CFPS Fellow

- 1975** – Fred Brooks “*The Mythical Man-Month*”
- 1978** – Putnam <estimation> Model
- 1978** – Alan Albrecht presents FP concepts IBM Guide/Share conference
- 1981** – Barry Boehm “Software Engineering Economics”
- 1983** – Albrecht / Gaffney “Software Function, Source Lines of Code and Development Effort Prediction: A Software Science Validation”
- 1984** – IBM published AD/M Productivity Measurement and Estimation Validation
- 1985** – Charles Symons Mark II function point method¹
- 1986** – IFPUG founded with Bill Hufschmidt as President. 1st conference held in Toronto, Canada.
- 1986** – Capers Jones introduces Feature Points
- 1986** – French Function Point Users Group (now ASSEMI)
- 1987** – IFPUG bylaws, Westerville, OH
- 1987** – Chris Kemerer, MIT An Empirical Validation of Software Cost Estimation Models
- 1988** – ASMA Australian Software Metrics Association
- 1989** – IFPUG CPM 1.0
- 1989** – Netherlands Function Point Users Group (now NESMA)
- 1989** – Watts Humphrey Managing the Software Process
- 1989** – UK Function Point Users Group (now UKSMA)
- 1990** – CMP 3.0 was Released
- 1990** – GUFPI-ISMA (Gruppo Utenti Function Point Italia)
- 1990** – The Australian Software Metrics Association (ASMA)
- 1992** – Italian Government adopts FP governance
- 1992** – FiSMA (Finland)
- 1993** – 1st CFPS certification exam

- 1993** – DASMA (Germany)
- 1994** – CMP 4.0 was Released
- 1994** – JFPUG (Japan)
- 1995** – IFPUG formalizes ISO/IEC JTC1 SC7 standards work
- 1997** – The International Software Benchmarking Standards Group (ISBSG)
- 1997** – AEMES (Spain)
- 1998** – ISO/IEC 14143-1:1998 Functional Size Measurement – Definition of Concepts
- 1998** – Capers Jones “Sizing up Software” in December issue of Scientific American
- 1998** – The COSMIC consortium
- 1998** – Brazilian Function Point Users Group (BFPUG)
- 2000** – Garmus/Herron “Function Point Analysis; Measurement Practices for Successful Software Projects”
- 2002** – IFPUG publishes hardcover textbook “IT Measurement – Practical Advice from the Experts”
- 2002** – Australia adopts Southern Scope
- 2003** – ISO/IEC 20926 IFPUG Functional Size Measurement Method
- 2005** – Mauricio Aguiar of Brazil becomes 1st first non-North American IFPUG President
- 2005** – Certified Software Measurement Specialist (CSMS) certification
- 2006** – Northern SCOPE™
- 2007** – SNAP (Software Non-functional Assessment Process)
- 2008** – Brazilian Government directive “IN04”
- 2009** – CPM 4.3.1 was made available
- 2010** – IFPUG approved for the Japan Industrial Standard
- 2012** – “The IFPUG Guide to IT and Software Measurement”
- 2012** – POSMA (Poland) founded
- 2015** – AMMS (Asociación Mexicana de Métricas de Software)
- 2016** – Malaysian Government adopts FP governance
- 2017** – IEEE PAR working group for SNAP
- 2017** – IFPUG Celebrates 30 years and the International Year of Software Measurement!

¹Function Point Analysis: Difficulties and Improvements, C. R. Symons, IEEE Transactions on Software Engineering, 1985.



**Q/P MANAGEMENT
GROUP, INC.**

Software Benchmarking including Agile

- **Worlds Largest and most accurate FP based database**
- **Measures productivity, effort, quality, schedule and more**
- **New Development, Enhancement and Maintenance**

www.qpmg.com
moreinfo@qpmg.com

***Providing Industry Leading Software
Measurement Consulting since 1990***

- **Certified FPA and SNAP Training**
- **Measurement and Estimation**
- **Productivity, Quality and Cost**

North America: +1 781.438.2692
Europe: +44 20.3287.921

When Metrics Mean Business

by **Mauricio Aguiar**

Abstract

Measurement specialists often wonder why so many Brazilian organizations use functional sizing as a basis for software development contracts. As a matter of fact organizations from several industries such as banks, airlines, telecommunication companies, and government agencies use function points as a basis for software development contracts. This article discusses the business drivers that have made Brazil the number one function point user in the world as well as the business processes involved.

1. Introduction

Having been founded in 1986, The International Function Point Users Group (IFPUG) is probably the oldest software measurement association in the world. Brazil is one of the top countries in IFPUG memberships, along with Italy, the U.S. and India. IFPUG certifies individuals that pass the IFPUG exam: Brazil is also one of the top countries in IFPUG certifications [1]. That makes Brazil one of the top countries in function point utilization. As a result several software measurement companies and independent consultants appeared in the Brazilian market in the last fifteen years. Even though there is no publicly available data, a single Brazilian company claims to count 60,000 function points per month [2]. At the cost of US\$ 1,000 per function point, that number could mean US\$ 60,000,000 changing hands each month based on function point counts performed by just one Brazilian software measurement organization [3].

In 1998 a group of Brazilians became IFPUG members and founded the Brazilian Function Point Users Group (BFPUG) that had a significant role in promoting software measurement and function point analysis in Brazil.

As more Brazilian organizations adopt the “price per function point” method, it is likely that more measurement-

related methods and techniques will become popular. For instance, the COCOMO II estimation model and the Practical Software & Systems Measurement framework are already used in Brazil; The Netherlands Software Metrics Users Association (NESMA) methods for early size estimation and enhancement counts are also used; the COSMIC measurement method is increasing its popularity, as well as the recently published IFPUG Software Non-functional Assessment Method (SNAP).

2. Use of Functional Sizing in Software Development Contracts

A typical software development project includes a request for proposal issued by the acquirer. Following that request, one or more suppliers present their proposals. These are usually based on an estimated amount of effort – a number of person-hours or person-months that constitutes the main input to the pricing process. Because the estimated amount of effort is highly dependent on the supplier’s resources and development process bids may not be easily evaluated by the acquirer. This pricing mechanism could be called “process-oriented pricing”, where the price is based on the inputs – the resources needed to complete the project. An alternate pricing mechanism could be designated “**result-oriented pricing**” where the price would be determined by the output, i.e., the amount of software delivered.

Process-oriented pricing, the more traditional way of pricing services, puts control in the hands of the supplier. This happens because the acquirer does not have the expertise or information to assess the supplier’s estimate. A simple analogy may illustrate this situation. Suppose you are on the road and your car breaks down. You manage to find a car repair shop nearby, where they agree to take a look at your car and tell you to come back after a couple of hours. When you come back the chief mechanic hands you a budget with several parts to be replaced and an estimated number of

A MESSAGE FROM PAST IFPUG PRESIDENT KRISTE LAWRENCE

Dear IFPUG Members, Congratulations on your 30th Anniversary! It has been my pleasure to meet you and to serve you as a member of the IFPUG Certification Committee and now as a member of the IFPUG Board of Directors. Each and every one of you helps to make IFPUG a special organization and I will gladly serve you in the years to come.

work hours. If you are not at least an amateur mechanic you will not be able to make a rational decision. In this situation the price is completely controlled by the supplier.

Suppliers initially tend to oppose the change to the new method, possibly for fear their profits will decrease. That may actually happen in the beginning, but as they learn to work with the method they realize their risks tend to be much lower than with fixed-price contracts, so they tend to become supporters of the “price per function point” way of doing business.

Functional size measures can serve as the basis for a result-oriented pricing method because they:

- are result-oriented by definition, as they measure the output (software) and not the input (work hours)
- can be understood and verified by both the acquirer and the supplier, bringing transparency to the business relationship
- can be standardized (IFPUG, MkII, COSMIC, and NESMA function points have become ISO standards [6])
- can be benchmarked (for example, see the ISBSG [7])
- can be used to manage the project scope [4, 5]

All those reasons have led Brazilian organizations to use functional size measures in software development contracts for new developments as well as enhancements. Acquirers aim to get control of the pricing process and ultimately pay lower average prices. They also want to be able to manage project scope and schedule, make sure budgets are not exceeded, and pay only for software actually required and delivered. Suppliers want to make sure every change request is adequately measured so that they get correctly paid. They want to guarantee that if the system size increases so does their revenue. If wisely used, functional size measures will support all those contract objectives.

2.1. Function Point Based Business Models

Brazilian organizations use functional sizing in software development contracts according to several business models

A simple definition of a business model is “a way of doing business”. The most common function point-based business models used in the Brazilian community are estimation models and pricing models.

Estimation models are predictive models used to calibrate a prescriptive model. Before using the “price per function point method” an organization will typically perform a study to determine the ideal price per function point to be paid the supplier. That type of study includes the definition of project

types, statistical analysis of historical and benchmark data, specific data collection, as well as the elicitation of business objectives. The outputs of such a study will be project types, productivity ranges, prices per function point, and/or possibly other measures of interest. Those numbers will be used in a prescriptive model that will ultimately be part of a contract. It should be noted that a contract for the development of a new software system will have different estimation requirements than an umbrella contract for an unknown number of enhancements to be performed over a period of years.

The most basic type of estimation model defines project effort as the product of size (in function points) and productivity (in hours per function point). Productivity is sometimes called delivery rate and usually varies with project type, being determined in a specific study as described above. Some more sophisticated estimation models – parametric models – define project effort as a function of size and several other factors. All estimation models use statistical techniques in their calculations. Some models like COCOMO II use more specialized techniques such as Delphi and Bayesian Statistics. Several models are implemented by tools such as SEER, SLiM, COSTAR, PRICE-S, KnowledgePLAN, Capers Jones’s SRM, and others.

In addition to contracts, estimation models are used in budgets, RFPs, make or buy studies, etc.

Pricing models are prescriptive models that establish the pricing and billing rules to be used in a software development contract. Those models can become quite complicated in contracts that deal with several types of service. While there are several model variations, only a few basic pricing models are used in Brazil, typically the Productivity-based Model, the Price-per-FP Model, and the Baseline-based Model.

The Productivity-based Model is based on the simple estimation model that defines effort as the product of size (FP) and productivity (Hours/FP). There will be a distinct productivity for each project type. Each new development or enhancement is sized in function points. The FP size is multiplied by the corresponding productivity giving the number of effort hours. The number of hours is multiplied by the contract price per hour giving the amount to be paid the developer.

The Price-per-FP Model simply assigns a different price per function point for each project type. Each new development or enhancement is sized in function points. The FP size is multiplied by the corresponding price per FP giving the amount to be paid the developer.

The Baseline-based Model assigns a price per function point to an installed application baseline. A fixed monthly fee is

(continued on next page)

(When Metrics Mean Business, continued from page 7)

charged for a service package, e.g. application maintenance and support. The price per function point is periodically updated as the application baseline grows. A service level agreement (SLA) handles all the details, including schedule, quality, incentives, penalties, etc.

Several other models are in use in Brazil, of which the following deserve to be mentioned:

Defect-based Model – This is a quality control model. A penalty is associated with a defect threshold, typically based on a defect density measure (defects per function point).

Negotiation-based Model – Even though this is a prescriptive model, it is negotiation-intensive. The supplier receives a request for proposal and is required to respond with the functional size, the number of hours, and the price estimated to complete the project. The acquirer receives the proposal and enters the FP size into an estimation model not necessarily known by the supplier. If the number of hours or price returned by the estimation model is greater than the number provided by the supplier, the acquirer accepts the proposal. Otherwise the acquirer rejects the proposal and starts a negotiation process to bring the number of hours or price to the level indicated by the estimation model. Estimation models used in this type of business model are not always objective. Some Brazilian acquirers use a customized COCOMO-like model in this manner.

Phase-based Model – Some acquirers outsource different parts (phases) of a project to different suppliers. For example, one supplier may develop the software product and other supplier may test it. In this business model each project phase is assigned a percentage of the total lifecycle, and the price per FP is divided accordingly. Each supplier is paid according to the defined project phase percentages and the project phases they are assigned to. Phase percentages are typically defined using historical data.

3. Technical Considerations Related to the Use of Functional Sizing in Software Development Contracts

Even though there are many benefits associated with the use of functional sizing in software development contracts there are still many issues to be resolved. Those have been addressed by Brazilian acquirers and suppliers both from the government and private sectors.

3.1. Predictive versus Prescriptive Models

One issue has to do with the difference between predictive and prescriptive models. Because most of the software development market still uses conventional effort-based pricing specialists tend to focus on predictive models to estimate effort. However, in Brazil the most important use of functional sizing is in pricing and billing. Even though there

are similarities between predictive and prescriptive models, there are also differences that must be considered before using an estimation technique as a basis for a pricing model. Both models share the goal of producing values that should be as close to the actual values as possible. Even though neither will be able to match the actual values exactly, both are expected to approximate them in the long run.

Estimating models are expected to give approximate results. A slight change in the inputs may not be reflected in the outputs. On the other hand, pricing models are expected to give exact results. A small change in the inputs should create a (hopefully small) change in the outputs.

Different estimators are expected to produce different results. A more skilled and more experienced estimator is expected to produce a better estimate than a novice, whereas an operator of a pricing model is expected to follow exact rules and produce exactly the same result as any other trained operator.

Estimating models often have subjective parameters. For example, the COCOMO II estimating model has parameters such as ACAP (Analyst Capability) and PCAP (Programmer Capability) with ratings 15%, 35%, 55%, 75%, and 90% [8]. An analyst or programmer team that falls in the 15% level is rated very low – at the estimator's discretion. Two distinct estimators could potentially disagree on those levels. Estimation models often have to be customized before they can be used for pricing purposes. Pricing models, on the other hand, have no room for ambiguity or subjectivity.

4. Benefits and Challenges of Using Functional Sizing in Software Development Contracts

The “price per function point” method potentially leads to better productivity and represents an improvement over previous effort-based methods. It brings transparency and objectivity to the negotiation process, being good for any application domain, development process, and technology.

Special care must be taken when determining the initial productivities in order to establish a balanced relationship between acquirer and supplier. While a good pricing model will reduce variation to an acceptable level, it is important to note that bad requirements do not favour accurate sizing. Poor requirements will increase the uncertainty in the sizing process. Most organizations will want to improve requirements before transitioning to the “price per function point” method. Any functional sizing method may be used, but it is highly recommended that measurement be performed or at least supervised by certified professionals. This will reduce differences in the interpretation of counting rules, especially between acquirer and supplier. Sizing may also be outsourced to a neutral third party organization in order to improve

transparency and minimize conflict. Non-functional items will continue to be a challenge until a non-functional measurement solution is found and accepted by the measurement community. Most of all, one should keep a win-win attitude and be aware that when using functional sizing in pricing models there will be gains and losses, but at the end of the day things will balance and everybody will win.

5. Conclusion

This paper has presented a short description of the utilization of software measurement in Brazil. After providing a historical perspective, several relevant topics were addressed, such as the difference between predictive and prescriptive models, and why the latter is so important in Brazil; why and how Brazilian organizations in the government and private sectors use functional sizing for estimating and pricing; the main technical difficulties that have been encountered and how they have been addressed.

References

- [1] IFPUG, unpublished internal reports, IFPUG, Princeton Junction, 2016.
- [2] Aguiar, Mauricio, "Domain-Oriented Modeling, Estimation and Improvement for Commercial and IT Systems Domains" (presentation), 26th International Forum on Systems, Software, and COCOMO Cost-Modeling", USC, Los Angeles, 2011, http://csse.usc.edu/csse/event/2011/COCOMO/presentations/Panel_Aguiar_2011-11-03.ppt
- [3] Jones, Capers, "Applied Software Measurement – Global Analysis of Productivity and Quality", 3rd Edition, McGraw-Hill, New York, 2008, pg. 310.
- [4] Government of Victoria, Australia, "southernSCOPE Reference Manual, Version 1", Government of Victoria, Victoria, Australia, 2000, <http://www.egov.vic.gov.au/pdfs/SSRefManual.PDF>
- [5] FiSMA, "northernSCOPE: customer-driven scope control for ICT projects" (brochure), FiSMA, <http://www.fisma.fi/wp-content/uploads/2008/09/northernscope-brochure-v152.pdf>
- [6] The International Organization for Standardization (ISO), website, <http://www.iso.org>
- [7] The International Software Benchmarking Standards Group, website, <http://www.isbsg.org>
- [8] Boehm, Barry et al., "Software Cost Estimation with COCOMO II", Prentice-Hall, New York, 2000.

About the author:

Mauricio Aguiar is President of TI Metricas, a Brazil-based software measurement company. Mauricio is an IFPUG Past President and currently serves as IFPUG Vice-President. He is a certified COSMIC Measurer and a Practical Software & Systems Measurement (PSM) Qualified Instructor, having received the "PSM Contribution Award" in 2010. Mauricio holds a degree in Engineering from the Federal University of Rio de Janeiro.

A MESSAGE FROM PAST IFPUG PRESIDENT MAURICIO AGUIAR

I have lots of memories from when I was IFPUG President from November, 2005 to October, 2007. The first thing I remember is Hurricane Katrina. The 2005 IFPUG Fall Conference was scheduled to happen in New Orleans, but due to the Katrina tragedy it was postponed to the following year, 2006. Needless to say, we were all very sad.

In the Spring of 2006 we had the first Functional Sizing Summit (FSS) in Cambridge, MA. The FSS was all about sizing, a very interesting and technical conference.

In the Fall of 2006 we had the first ISMA in San Diego. I'm proud that ISMA is still alive to date. What started in San Diego has already been in Madrid, Rome, Sao Paulo, Rio and Mumbai - ISMA is certainly bound to happen in many more international venues.

Some of you who went to San Diego may remember the Board and committee chairs had a strategic planning session with Marilyn Bushey in September of 2006. That may have helped to bring a common understanding and shape our future.

The last memory I have from my Presidency is signing the contract with Prometric in October of 2007. That would automate the CFPS exam and make it available to many people in several countries.

An Interview with Capers Jones

by David Herron

Capers Jones is currently the Vice President and CTO of Namcook Analytics LLC. He is also the founder and former chairman of Software Productivity Research LLC (SPR). Capers Jones founded SPR in 1984. Before founding SPR Capers was Assistant Director of Programming Technology for the ITT Corporation at the Programming Technology Center in Stratford, Connecticut. He was also a manager and researcher at IBM in California. Capers is a well-known author and international public speaker. Some of his books have been translated into six languages. All of his books are translated into Japanese and his newest books are available in Chinese editions as well.

I have known Capers for over 25 years. Most recently we had a chance to catch up and I took the opportunity to ask him a few questions.

Why is it important for organizations to consider measuring their software?

Well, software is one of the most important intellectual assets in the modern world. It controls almost everything; cars, computers, houses, security, banks. You really need to know quality and productivity and security for software because it is the dominate driving force

of industry. And unfortunately some of the older metrics like lines of code and cost for defects actually conceal economic value and they conceal bugs and quality and everything else and so function points are the only thing that give you a good view of what really is happening with the software.

More than once you have spoken out against a LOC measure.

I have. Lines of code penalizes high level languages, it makes Assembly look better than Smalltalk and modern languages and Ruby. They also can't measure requirements, design and non-coding work which is over half the total effort.

What would you say to the organization just starting to consider software measurement as to why they should be using function point based metrics?

Well there are many many variables that affect software; programing languages, team experience, nature of the methodology, the size of the application. Function points enable you to look at these individual factors separately and then to look at the overall aggregate impact of those factors. And that's really

what you need. You need to understand everything that impacts software and function points is the best metric for doing that. What I like to do is give actual numbers to clients to give them a context of what it really means. Like if your productivity is below five function points per month you got something wrong. If it's above eight function point per month you're doing pretty well. If you've got more than four and a half bugs per function point you've got too many bugs. If you've got less than three you're doing pretty well on quality. If you're removing less than 95 percent of the bugs your quality control is poor. If you're removing more than 98 percent of the bugs then you're doing a good job. So function points combined with defect removal efficiency gives you a good quality handle. And then function points per month or work hours per function point gives you a good productivity handle.

If you had to pick one a key metric what would it be?

Well it would probably be work hours per function point. The problem with function points per month is that in India they work 190 hours per month. The Netherlands they work 115 hours per month. So if we use function points per

A MESSAGE FROM PAST IFPUG PRESIDENT SCOTT GOLDFARB

Happy Anniversary IFPUG! 30 years serving IT professionals is a great accomplishment. It was an honor serving as IFPUG President and I'm very pleased I was able to contribute to the success of the organization. Serving on the Board of Directors gave me an opportunity to give back to an organization that helped me develop as a software measurement professional. In addition, I will always treasure the personal and professional relationships that developed over the years. I encourage everyone to get more involved and take advantage of the benefits of membership.

month you have to adjust for local work hours, on the other hand work hours per function point is valid in every country from every application. Plus you can use that to measure work hours per function point for requirements, for design, for coding and for testing. You can look at the activities as well as the total project. So work hours per function point gives you a metric that is valid internationally in every country and valid for every activity including project management.

I have also known you to be a strong advocate for defects per function point.

Well yes. They are very good for that. I measure requirements bugs per function point, design bugs per function point, coding bugs per function point, documentation bugs per functions and bad fixes. A bad fix is a fresh bug introduced while trying to fix an older bug. The U.S. averages about 7 percent of bug repairs have defects in them. The current U.S. average for bugs per function point is 4.25 bugs per function point. The current average for removing bugs before release is 92.5 percent, but the best in class are topping 99 percent. They are the ones that use inspections and static analysis and about 10 kinds of testing.

I know you're a proponent of expanding function point logic into other areas. What would some of those areas include?

Well software is not the only thing we build. It would be nice to have hardware function points so you could do integrated cost studies on products like computers and servers that have software and hardware mixed in together.

Plus most companies own more data than they own software today. We need a data point metric because the cost

of data quality is expensive. Data bugs are troublesome and there's no good size metric for data so I think we need a data point metric. The combination of hardware function points and software function points and data function points would open up a lot of important economic studies that we can't do today. Plus we should also look at value in terms of function points. Financial value is easy to measure, you know, in dollars and cents. But intangible value like the medical value of a new kind of therapy or the defense value on a new kind of weapon system, those are not dollars and cents values those are intangible values. So I think we need a value point metric that allows us to compare function point cost against function point value.

What organization would be best suited to pursuing some of these unique measures.

It would be nice to say that IFPUG or one of the function point groups like COSMIC would do that work but they don't really seem to be set up for actual research. So the next tier would probably be universities because they do research. But I think large corporations like IBM (where function points were developed) and Microsoft, they would probably do the work. Still there are other companies, Microsoft, Apple Computer, Computer Aid any of the big software corporations could do that work.

I would be interested in your opinion on SNAP.

Nonfunctional requirements are important and they are troublesome and expensive. But my only wish is that the SNAP metric should have been made additive and mathematically consistent with function points. So if you have

something like a 1000 function points and 200 SNAP points you can add them together and end up with the total of 1200 points. But, because they are additive that means you need to do double cost estimates. And the only place the numbers come together is in the final financial value but it is not very effective and it is counter to the way the rest of the world measures things. So I think SNAP should have been made numerically equivalent with function points.

People are starting to use SNAP but it's awkward to use. It doesn't fit into any of the standard cost estimating tools yet.

What countries are making the best use of their software data?

Well I would say Brazil is. Starting in 2008 the government of Brazil mandated function points for all government contracts so they're looking pretty good. Japan is looking good. You know the Japanese function point group is a major organization too and they've got good work going. South Korea has mandated function points for government contracts. Malaysia has just started. So I would say that the Asian countries of Japan, South Korea and Malaysia are looking good. Also Brazil and some of the South American countries like Mexico are looking good. The countries that should be looking good but haven't done much are China and Russia.

Has the use a function pointer over the last 10 years increased or decreased in the U.S., Internationally.

I know of 25 countries where function point usage is increasing. So overall I'd say function point use is increasing but one issue is that because Agile is so popular and Agile uses story points. So we have some dilution with story

(continued on next page)

(An Interview with Capers Jones, continued from page 11)

point metrics which are really terrible. They're not standardized and there is no ISO standard. They vary by 300 or 400 percent. So the Agile people are pushing back against function points and trying to use story points which don't even work. So Agile is pushing in the wrong direction, but it's very popular.

What do you see currently out there in terms of software measurement automation?

Well, all of the parametric estimation tools like COCOMO, Seer, Slim, Price S and Software Risk Master, are much more accurate than manual estimates for big applications. I did a study of side by side comparison of 50 manual estimates and 50 automated estimates and the average error for the manual estimates was 34 percent. The average error for the parametric estimate was only about 6 percent. So below 500 function points they are pretty equal when you start getting into bigger systems, (2, 3, 4, 5 thousand function points) parametric

stay within 5 or 6 percent and the manual estimates go off the scale 40 or 50 percent optimistic.

So automated estimating tools are good. We have an automated sizing tool that predicts size for any application in about five minutes or less. (More information about this tool can be found at <http://www.namcook.com/>) There is a lot of good data on our site as well as information about our sizing tool.

You've worked with a number of prominent individuals in the software industry who were some of the more memorable individuals.

I have been lucky to meet a lot of them. Of course a lot of the people that were pioneers in metrics are getting older. We are all in our 70s and 80s and of course I met them when they were much younger. For example I met Fred Brooks (Mythical Man Month) we worked for IBM at the same time. Barry Boehm (COCOMO model). We're about

the same age and we've spoken the same conferences and know each other for a long time. Jerry Weinberg the developer of the psychology of computer program and I met Alan Albrecht's right after he invented function points we both spoke at the same conference that IBM hosted in 1978. I had a paper on the problem of lines of code, but I didn't know how to solve it and he had a paper on function points and he did know how to solve it so we became personal friends. Not only did we become friends but he came to work for us at Software Productivity research. And I know Watts Humphrey, we've worked together. Watts was at IBM at the same time and we stayed friends. I know a lot of the pioneers in many aspects of software over and above metric software.

Capers, thank you so much for sharing this time with me. We have known each other a long time during which you have been a role model, a mentor and a good friend.

A MESSAGE FROM PAST IFPUG PRESIDENT JOE SCHOFIELD

Fortunately, my term as IFPUG President began when I was retiring from Sandia National Labs after 31 years. Less fortunately, my term as IFPUG President began as I embarked on a new life as an independent consultant with a focus on IT governance, process transformation, measurements and objectives, agile coach and certification trainer. Yet my colleagues on the Board and throughout IFPUG enabled us to accomplish several novel endeavors which, are chronicled in earlier MetricViews articles. Thematically, we invested in ourselves and put our cash reserves to work on our behalf.

Personally, I was elated to work with so many professionals throughout IFPUG during my seven years on the Board—all without mutiny or being voted off the Board. Meeting our fellow IFPUG members in Brazil, Canada, Spain, Poland, and in the more traditional US venues was an honor I'll not forget. Our engagements with other measurement organizations elevated the community as a whole. Few folks realize how long and arduously most Board members labor to make the difficult look mundane, and that description extends to so many of our committee members, volunteers, and CMA as well. Thank you all.

Our special events were always that – special. As an organization we celebrated. We experienced joy. We prompted occasional tears. We made new friends. We lost some dear ones too. We took measurement into new frontiers, as if it was a SNAP. We recognized a few Fellows. In all we made the world a more credible place for software measurement—we promoted the “international” in IFPUG.

Those are but a few of my IFPUG memories. All of these can be summarized as making and keeping “old friends.” Bless ya.

Thirty Years of IFPUG!

by Paul Radford

Thirty years. It's a long time in IT. An organisation set up to formalise a software estimation technique back in the glory days of Cobol is still relevant today. Underpinning technology has changed massively, as have the political and business conditions under which we strive.

Despite ourselves, the basic need for software measurement will always be with us. At some point, someone needs to know. And IFPUG has established a long term structure to provide legitimacy and a discussion point for this form of measurement and its uses.

30 years longevity is a rare achievement in the fast moving world of information technology. I well remember attending vast, glamorous conference events in Vegas on technology that came and went in the space of five years – whilst IFPUG is still with us.

But just growing old, whilst better than the alternative, is no great achievement of itself. It is what IFPUG has achieved – and may yet achieve in the future – that makes it important and relevant.

The birth of IFPUG – a formal structure to support a specific form of analysis – was the result of real vision. The understanding that software was so integral to all our futures and that measurement of that product is a significant step forward was a key to that vision. Definitions and guidelines were established, based on Albrecht and his team's original work at IBM.

Although really a USA consultants and users group at inception, IFPUG provided a focus for a growing worldwide tide of IT professionals who were a little tired of being blamed for continual cost blowouts and software implementation delays. They wanted a bit more than “gut feel” to support forecasts – they wanted to know their productivity and how it was impacted. Scope creep was poorly understood and even less well controlled. The usual solution for any under-performing project (against an original budget guess) was to throw out the current people, then throw large buckets of money at the project which would then be micro-managed to a long, slow and sometimes successful conclusion which pleased no-one.

Breath of Hope

Albrecht's function points were a sudden breath of hope. Studies in the UK, Germany, Australia and the USA all backed up a positive correlation of function points to effort. Not a direct correlation, but superior to any other measurable attribute available prior to actual commencement of build. Or after the build.

The existence of IFPUG inspired the creation of groups around the world, including the Australian Software Metrics

Association (ASMA) in 1988, from where key products and further initiatives followed (e.g. Function Point WORKBENCH, ISBSG).

It was the consistency of an after-the-build capability which caught the eye of management. For larger organisations, software always has been a factory. Monitoring productivity to justify spending is inevitable. IFPUG based function point analysis was mooted to be able to provide that basis. Huge software measurement programs sprung up all over the USA. The imprimatur of IFPUG became a key factor in ensuring, as far as practicable, that analysis and function point assessment was performed in a consistent manner across organisations and across time.

IFPUG Conferences were getting bigger – and running twice a year.

In 1996, IFPUG went truly international with a Rome conference and a galaxy of stars line-up of speakers. Metrics professionals from all over the world attended – connections were made, ideas exchanged.

But success always brings difficulties.

Despite the networking success of the Rome conference, it was a financial disaster. Consequently, for the next decade all conferences were held in North America.

And there was pressure on the counting rules and how they were applied. Now measurement of IT was possible, senior management demanded proven results of the investments they were constantly making in new technology. Although measured by people following the rules of IFPUG, results varied greatly according to interpretation. And when push came to shove in a legal sense, the often illogical and irrational interpretations of Value Adjustment Factor guidelines became critical. On the other hand, for those who had invested heavily in establishing data under these guidelines, radical change was certainly not desired.

IFPUG was finding that mastering a standard to meet all needs somewhat of a challenge.

Standards

Part of this problem was the basic flaw in applying all of Albrecht's initial findings to a general standard. His purpose was an estimating tool for one company, not a sizing standard. The definition of end user functions and its positive relationship with effort and cost was a breakthrough. However, the concept of the Value Adjustment Factor was fundamentally flawed. And IFPUG picked up some idiosyncrasies of its own (e.g. the physical act of pressing the Enter key was decided

(continued on next page)

(Thirty Years of IFPUG, continued from page 13)

to be equivalent to one logical element). At one point, every single error message was to be counted as a function – an opportunity for some developers to inflate their productivity incrementally. Through a long period, IFPUG strived to meet twin masters of repeatability and practical utility. And still does. The emerging hybrid has simply grown bigger and more complex, until what was once a relatively simple technique now even requires 3rd normal form data analysis as a pre-requisite activity.

But back to the early days. As IFPUG was loathe to admit to any flaws nor submit to change, a wide number of alternative sizing philosophies emerged. As still exists today, universal agreement as to how measurement should be done was not practicable. Perhaps more importantly, whilst measurement professionals argued, the concentration of the IT community was moved to a focus on process. Improved process would automatically lead to better outcomes. Measurement that reinforced this paradigm was well received; measurement that challenged this paradigm was simply unwelcome.

All these enhanced – and sometimes cumbersome – processes eventually became so overwhelming that a demand for a more responsive approach led to various forms of Agile. Both consumers and developers were enjoying the interaction and were loathe to risk comparative measurement. The nature of Agile may sometimes lead to more re-work in the initial build – although this may simply replace re-work that would occur later under a different approach. But explaining such complexities is best avoided by simply avoiding measurement in the first place.

And still measurement professionals around the world argued and went their separate ways.

Outsourcing

The 21st century brought mass outsourcing of software development and support. This provided yet another opportunity where measurement was desperately important. Whilst large numbers of apparently highly qualified people working for very low rates seemed a good deal – was it really? Some German studies indicated that there were serious deficiencies in the model which often led to excessive cost. But decisions to outsource large IT investments are made at a high managerial level. Costs are immediately cut – no-one is seriously interested in learning if this truly is efficient and nor do they care if productivity is poor. It is cheaper on the books and it is someone else's problem. Measurement beyond this can only bring embarrassment.

But the world moves on – and the inexorable hand of measurement moves with it. Just as early centres of metrics

excitement sprang up in the USA as well as The Netherlands, Germany, Australia and Italy, new centres of interest keep arising. And new managers seek to highlight how much prior poor decisions in outsourcing have cost them. Brazil established a significant interest in and use of metrics – whilst India has a nascent base of interest which simply needs appropriate nurture. And, these days, IFPUG is keen to nurture and to share in ways that were not possible in the past. Returning to a conference in Rome was a return to an attitude of supporting and sharing.

The fact is: most surviving methods of sizing software are largely based on the IFPUG definition of a user function. Some have varied this slightly; perhaps even improved it. After this initial analysis, means of allocating weights and numbers can differ markedly. However, results tend to be remarkably similar.

Future Vision

IFPUG needs to find a way forward to re-establish a paradigm that can be embraced worldwide – one that will enable measurement discipline to start confidently meeting the needs of IT management. IFPUG is currently anchored with several hundred pages of daunting guidelines that it is too scared to touch lest it offend one of its many forms of users. A better framework is needed.

But IFPUG appears now to be the organization to take software measurement forward. Conservatism has been replaced by a spirit that is both inclusive and communicative. There is still much listening to be done – but IFPUG has demonstrated long term resilience and the ability to rally people to the importance of this work.

It may be a little like hoping the United Nations could actually work – a touch romantic. But, like the UN, much good can be achieved on the journey.

But it is well to remember that IFPUG has achieved all it has based significantly on volunteers. Many people have given willingly and generously of their time and their attention in order to facilitate the work of IFPUG. Some of them are referenced in this edition but there are many others whose interest, intelligence and skill have contributed greatly to the ability of IFPUG to serve the IT community. It's achievements and its approach is greatly influenced by those individuals. In some cases they take the blame for IFPUG's not inconsiderable flaws – but what it has achieved is both remarkable and enduring.

About the author:

Paul Radford, Owner of Charismatek Software Metrics

Simple Function Points

by Roberto Meli

The functional size measurement is now more than 40 years old and has entered maturity and a relative stability. Unfortunately its usage is not yet spread as it “could” and “should” be in the market. The reasons are basically four: 1) the high level of knowledge and professionalism needed to master the most diffused methods; 2) the high level of requirements’ details and time needed to measure according to the standard rules and practices (described in quite big reference manuals); 3) the progressive uncoupling between the user’s functionalities provided by the development and maintenance projects (measured as “released FP”) and the software artefacts that must be worked - realized or modified - in order to deliver those functionalities; 4) the increasing importance of non-functional requirements in determining the final cost of a software project.

The third and forth issues are not dependent on a specific Functional Size Measurement Method and may be faced using guidelines and new metrics. I will not deal with them in this article. The first and the second issues were at the origin of the definition of a new measurement method, derived from the IFPUG standard method and called Simple Function Points (SiFP).

Today there is a great attention to the “functional size approximation” subject in order to improve the acceptability of the methods to the technical community and to give an answer to the first two issues. This is of course important and I remember the high interest around the presentation of the Early & Quick Function Point technique at the IFPUG annual conference in Scottsdale (Arizona) in the far 1997! Unfortunately we do not only need an approximation method to be sustainable (low cost, easy to learn and use, early in the life cycle, using less documentation and requirements details) but the business community is seeking for a “measurement” method (with precise results) having those characteristics.

In fact, FPA calls for quite detailed descriptions of requirements, which are often not available in the early phases of development, when measures are needed for effort estimation. Even when detailed requirements are available, standard FPA involves a quite thorough analysis of requirements: this takes time, so that measures may not be available when needed.

The IFPUG method has been a positive revolution in the way in which to measure software and it has been a lighthouse in the darkness for at least 3 decades ! The evolutions in the versions of the method were significant till the 4.1 and then became “fine tuning” till now. The times are mature for an evolution in the direction of “simplification” and agility. As it happens very often it is very difficult to change direction when you have invested a lot in assets (specialists, certifications, tools, baselines etc.). Nevertheless we are in a business situation where “innovation” is a key success factor to stay alive and improve. There are two main assumptions that have prevented the simplification of the standard IFPUG approach till now. Both them were impossible to demonstrate as being true or false till the availability of large benchmarking data bases like ISBSG. Unfortunately those assumptions were also very intuitive, leading the practitioner to feel comfortable with them.

The assumptions that every FP practitioner has always made is that the internal details (complexity rules based on DET, RET, FTR) and types of elements (EI,EO,EQ,ILF,EIF) are indispensable 1) to represent functional “value” to the user and 2) to better correlate functional size to the effort in a cost model. The research made on these two assumptions using international data bases has empirically demonstrated that they are not true. This is not the right place to bother the reader with statistical analysis details but the bibliography is linking the appropriated information sources (thanks to Prof. Luigi Lavazza to have conducted very robust and clear statistical analysis). The Simple Function Point (SiFP) method was

(continued on next page)

A MESSAGE FROM PAST IFPUG PRESIDENT DAVID GARMUS

Being elected as the President of IFPUG was the fulfillment of a career in software measurement. IFPUG enjoyed many achievements during my term, but the agreement by Addison Wesley to publish our first IFPUG book on IT Measurement was the most memorable success. Conversely, it was the year of 911, and our attendance at the IFPUG Fall Conference suffered greatly as a result...truly a year to be remembered.

(Simple Function Points, continued from page 15)

derived from the IFPUG model but uses only two types of elements (logical elementary transactions and logical data store) and does not consider internal details to assign points. For this reason the method is dramatically simpler than the classical one and it is easily usable by any team member with a small training investment. Surprisingly the correlation between SiFP and IFPUG is impressively high and the ratio between the two measurements is 1/1 with a high statistical evidence. Effort models built on SiFP have the same precision as those built on IFPUG. Any IFPUG expert is automatically a SiFP expert, since the second method is a “subset” of the first one. CFPS and CFPP certifications may be considered valid also for SiFP. The training duration is 1/3 of the usual standard training and the reference manual is only 24 pages long. Measurement may be spread to the analyst community with much higher success rate. SiFP may be considered an agile measurement method that requires a time to be computed which is compliant with the short iterations of development cycles in Agile frameworks and the level of detail of information and documentation needed is minimal. SiFP is not only an IFPUG approximation method but it must be considered as a measurement method highly convertible with IFPUG method. Converting the existing baselines is a matter of minutes if the counting details are available (number of EI,EO,EQ and number of ILF, EIF), a spreadsheet is enough and it is not needed to come back to the requirements documents. SiFP is much more easily automated starting from the code giving a better compliance between human and automatic measurement. A measurement expert maintains her/his role in applying the standard rules to the different contexts, frameworks, processes, documentation standards, cost models etc. What is eliminated is the boring part of the game... The published literature support all the findings here reported.

These findings may lead to the question: why do we need to maintain an approach which was based on assumptions that

have proven to be not representative of the reality ? The “sliding doors” are opening to a confluence of the simpler method into the traditional one as an evolutionary step, revamping a stable approach with innovation and a stronger suitability to the market demands in order to maintain a lighthouse role to IFPUG. “Keep it simple” is a must !

References

1. Standish Group 2015 Chaos Report - Q&A with Jennifer Lynch, <https://www.infoq.com/articles/standish-chaos-2015>
2. Standish Group 2015 Chaos Report - Q&A with Jennifer Lynch, <https://www.infoq.com/articles/standish-chaos-2015>
3. Quantitative Software Management (QSM) Software Sizing Infographic outlines how and when to use functional size measurement during the software development life cycle. Both authors were involved in the development of this important infographic. - <http://www.qsm.com/infographic/software-sizing-matters>

About the author:

Roberto Meli graduated in Computer Science in 1984. During the past 30 years he has developed focused competences in project management and software measurement areas and has written more than 75 papers for technical magazines and international conferences. He is a consultant and lecturer in training courses on project management and software measurement for many major Italian companies and public organizations. He developed the Simple Function Point method, the Early & Quick Function Point Analysis method. Currently, he is President of the Simple Function Point Association (SiFPA – www.SiFPA.org).

A MESSAGE FROM IFPUG PRESIDENT TOM CAGLEY

The role of the President of IFPUG, twice, has been a highlight of my professional career. Over the years I have been involved with IFPUG, we have worked diligently to be truly international participating in conferences in Italy and India. The people of IFPUG are what makes participating in IFPUG special, the relationships have made my career much richer and I hope that I have contributed even a portion of the benefit you have provided to me! — Happy Anniversary!

Remembering Allan Albrecht

by David Herron

I have had the pleasure of being involved with IFPUG for the past 27 years. During that time I have had the good fortune to meet and work with many wonderful people in the software industry including Capers Jones and Ed Yourdon. But one of my most cherished relationships I formed was with Alan Albrecht.

Alan Albrecht has long been credited with being the 'father' or inventor of function points. Truth be known, he was part of team of IBMers who were working to find a more effective way to accurately estimate software projects. However, it was Alan who first introduced function point analysis at an IBM conference and from that point forward his name was associated with function points.

When I first met Alan I was a new employee at Capers Jones' firm, Software Productivity Research. I was working alongside Scott Goldfarb and we were responsible for providing consulting and training services to SPR clients. One day, Capers came to me and asked if I would like to teach a class on function points to a group over in London. Of course I said yes, but with the caveat that I had never traveled outside of the country and more importantly, I had no idea what a function point was!

Capers was reassuring and told me that I would be quickly educated on the subject by his longtime friend from their IBM days, Alan Albrecht.

Alan and I spent the next several weeks pulling together materials that would make up what we thought to be one of the first publicly available classes on function point analysis. Naturally along the way, not only were we developing the training materials, but Alan was providing me with an education on function points. I have always been grateful and honored to have been trained by one of the founding fathers of function points.

All of this took place back in the late 80s and early 90s. The software environment was less complicated with regard to languages and technologies being used to design, develop and deliver software. Alan's approach to counting function points



was very straight forward and he always had a clear and concise answer for any nuanced situation that was raised in class.

We spent the next two years 'touring' the U.S. and presenting our class to SPR clients. It felt as though we were spreading the gospel on software measurement.

Spending that much time with any one individual doesn't necessarily mean you will become close friends but in our case, that is exactly what happened. Even though Alan was somewhat older than I was we had numerous things in common, sailing being one of those activities that we both

enjoyed. Alan was retired and living on Cape Cod and so we spent many a weekend sailing the waters of Cape Cod bay.

Allan J. Albrecht, the father of Function Points, passed away in 2010. I was blessed to have had that time to learn from Alan not only the function point counting method but also to learn from the wisdom and experience of someone older and wiser. Alan, along with Capers Jones, inspired me and my business partner, David Garmus, to write two books on the subject of function points and software measurement. I like to think that in some small way, we carried on the legacy of Alan Albrecht through our teachings and our writings.

About the author:

Mr. Herron was one of the co-founders of David Consulting Group. Over the course of his professional career Mr. Herron has provided consulting, education and coaching services for a variety of IT organizations. He is an acknowledged authority, instructor and lecturer in the areas of function point analysis, project estimating, performance measurement, process improvement and organizational change management. His books, coauthored with D. Garmus, are to this day widely accepted as the noted authority on the topic of FPA and software measurement.

A Look Back at 30 Years



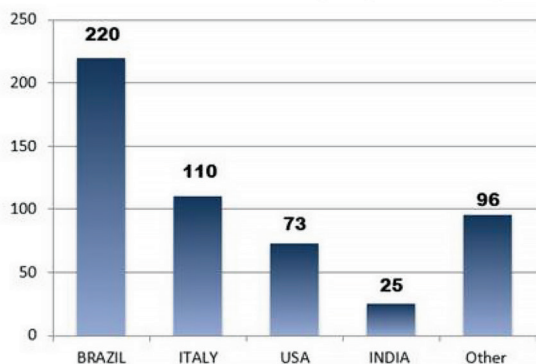


IFPUG By The Numbers

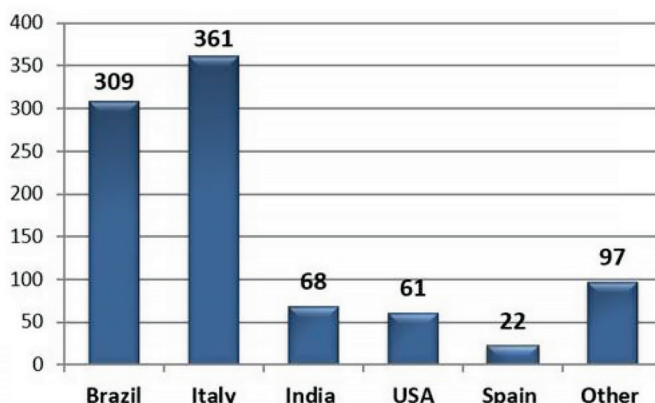
Number of CFPS

918

Membership by Country



Number of CFPS by Countries



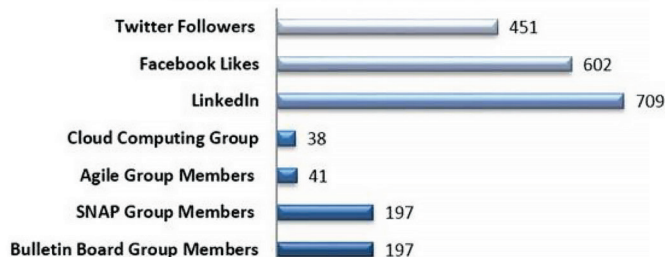
Number of Certified/Extensions



May 2017 Certification Totals

Number of Social Media Followers

ISMA INSIGHTS & SOCIAL GROUPS



A MESSAGE FROM PAST IFPUG PRESIDENT FRANK MAZZUCCO

What I remember most from my time as IFPUG President was about when we started to become a truly international organization. That was just a beginning, of course, and since then the leadership and members have done a tremendous job of expanding on that so that we do truly serve the whole world.

I have always been proud to be part of an organization dedicated to improving software processes via measurement. My challenge to future leadership—and to the members as a whole—is to continue to act as an advocate for software measurement in your own organization and in the software community as a whole, so that your management will continue to recognize its importance.

CELEBRATING THE INTERNATIONAL YEAR OF SOFTWARE MEASUREMENT

Nesma Congratulates IFPUG on Their 30th Anniversary

Submitted by Eric Der Vliet

In this contribution for IFPUG's MetricsView, NESMA first congratulates IFPUG with its 30th anniversary.

NESMA is an independent international organization that focuses on software metrics and software measurement and that celebrated its 25th anniversary in 2014. Till 2014, NESMA used to be an acronym for Netherlands Software Metrics users Association. After a reorganization NESMA is used as a brand and not as an acronym any more. Resulting from this reorganization, NESMA has chosen to adopt the slogan: 'NESMA, more than just points' and adopted the url www.morethanjustpoints.org.

Sizing (e.g. function points or other size measures) remains an important aspect and NESMA remains the provider of one of the (ISO/IEC) certified standards for functional size measurement of software. However, the strategy of NESMA is to provide objective and independent information on the use of software metrics in business areas like software cost estimation, benchmarking, outsourcing, productivity measurement and project control.

The change in strategy is one of the reasons that NESMA collaborates with ICEAA in the development of the 'software Cost Estimation Body of Knowledge' (sCEBoK). The objective of this program is to provide a certification for software cost estimation and to bring cost estimation for software to a next level. The first 7 modules of the sCEBoK were presented during the ICEAA workshop in Portland in 2017.

The wider focus of NESMA is also reflected in the

working groups and deliverables, like the Framework for Estimating Packaged Software and the working group Metrics in Contracts that developed several mini guides and is currently working on new deliverables.

NESMA changed with this new strategy its focus from the Netherlands to an international audience and works together with international organizations like ISBSG and Leda and also with other international sizing organizations like IFPUG and Cosmic. NESMA participates in the IFPUG counting practice and works together with IFPUG and Cosmic on the marketing of functional size measurement. In 2016 NESMA started a collaboration with SPI China to certificate members of SPI China in the NESMA method and deployment of the NESMA standard and deliverables. We are proud that in 2017 the first Chinese candidates passed the NESMA CFPA exam.

With this change in strategy, NESMA also changed its organization in a board that focusses on the strategy and practices that focus on the operation. This makes the organization more effective and more prepared for the future. A future where the focus will remain on 'More than just points' and where NESMA looks forward to continue its collaboration with other international organizations to bring cost estimation across the globe to a higher level and make together this 'International Year of Software Measurement' successful.

For more information please visit us on our website www.nesma.org or contact us at office@nesma.org.

IFPUG Board of Directors

Tom Cagley, President

Premios
t.cagley@premiosgroup.com

Mauricio Aguiar, Vice President

TI Metricas
mauricio@metricas.com.br

Christine Green, Secretary, Director of Certifications

Debra Maschino, Treasurer

NASCO
debra.maschino@nasco.com

Kriste Lawrence, Immediate Past President

DXC Technology
kriste.lawrence@dx.com

Carol Dekkers, Director of Communications & Marketing

Quality Plus Technologies Inc.
dekkers@qualityplustech.com

Dácil Castelo, Director of Sizing Standards

Leda-mc
dcastelo@leda-mc.com

Pierre Almén, Director of International Membership Committee

ImproveIT
Pierrea@coolmail.se

Luigi Buglione, Director of Education & Conference Services

Engineering.IT SpA
luigi.buglione@eng.it

The COSMIC Functional Size Measurement Method

Submitted by Charles Symons

The 'Common Software Measurement International Consortium' (COSMIC) congratulates IFPUG on its 30th Anniversary.

COSMIC was formed in 1998 by an international group of software metrics experts to develop a new-generation software sizing method:

- based on fundamental software engineering principles;
- applicable to business, real-time and infrastructure software, at any level of decomposition from whole systems down to re-usable components (or single User Stories);
- independent of the technology or processes used to develop the software (however, measured sizes should obviously correlate well with development effort);
- 'open' with all documentation, including the basic 'Measurement Manual', available for free.

The method's principles were first published in 2000 and remain valid to this day, so the method is stable and 'future-proof'. However, since 2000, we have made great progress in many areas.

- We have published specialist Guidelines with many examples on how to apply COSMIC sizing to business and real-time applications, data warehouse and SOA software, and for use in Agile projects. We also have many case studies, ranging from sizing mobile apps and web software to domestic appliance (real-time) software.
- Another Guideline describes several approaches to measuring approximate COSMIC sizes, e.g. when requirements are still evolving early in the life of a project.
- Many users of the method have published data showing excellent correlation of COSMIC sizes with effort in domains such as business and web applications, real-time embedded software, Agile User Stories.

- Studies have shown that Non-functional Requirements often evolve as a project progresses, wholly or partly into functionality that the COSMIC method can measure. (Examples: a NFR for 'usability' may evolve into a requirement for GUI functionality. A 'maintainability' NFR may lead to a requirement for parameter tables.) Using COSMIC, there is no need for a separate size measure for NFR.

- A Guideline advises on how to convert Function Point sizes to COSMIC sizes.

- The method's basis on fundamental software engineering principles means that tools are starting to appear for automatic COSMIC size measurement, e.g. of requirements expressed in UML Use Cases, in Matlab Simulink tools, the SCADE safety-critical language, and from static and executing Java code.

All COSMIC documentation and more information on all of the above is available for free-download from www.cosmic-sizing.org. The method has been accepted as an ISO/IEC standard (19761).

The world of software metrics and estimating methods has many players, but together we reach only a small part of the software community. COSMIC believes that we need to collaborate more if we are to grow our collective market presence.

COSMIC and IFPUG have already jointly produced the high-quality Glossary of NFR and Project Requirements terms. We are currently again working together, also with Nesma, to produce papers aimed at increasing the acceptance of our methods by the Agile community. All three bodies are working together with the ICAEEW on a syllabus for certification examinations for software cost-estimators.

We look forward to collaborating with IFPUG and other Software Measurement Associations through the International Year of Software Measurement and for many more years.

The International Software Benchmarking Standards Group (ISBSG) Congratulates IFPUG

Submitted by Harold van Herringen

ISBSG would like to congratulate the International Function Point Users group on their 30th anniversary.

The ISBSG is a not-for-profit organization, based in Melbourne (Australia) that collects software data from the industry with the goal to help management in the IT industry to improve decision making based on relevant data instead of (expert) opinions.

IFPUG function points is the most commonly used sizing method. It is better suited to applications that contain user interface, reports and database. This is the majority of applications in the ISBSG data repository.

Currently, 2 repositories are available: Development & Enhancements (just released a new version, now over 8000 projects) and Maintenance & Support (over 1100 applications). The ISBSG data can be used for software project estimation, reality checks of proposals, project control, scope management, (supplier) performance measurement, benchmarking and research. ISBSG regularly analyzes the data and publishes special analysis reports on this analysis. ISBSG data helps you to make better decisions, based on objective data instead of subjective opinions.

Important factors of the size measure are:

- Independent certified measurers should calculate the same size for an application
- The size is able to be measured at any stage of the project
- The size can be used in contracts as a basis for costing and payment

In addition to IFPUG Function Points, ISBSG also stores the following methods for sizing software:

COSMIC – useful for complex real-time applications.

Feature Points – developed by Capers Jones and is an extension of FPA to count real-time applications. Not widely used.

FiSMA – used by Finland Software Metrics Association.

LOC – not very reliable and not really used anymore.

Mark II – used in United Kingdom.

Nesma – used by Nederlands Software Metrics Association.

ISBCG Challenge: How to stay relevant in an Agile world

Nowadays, many organizations move from a traditional software development method to Agile and DevOps team approaches. Although these modern ways of software development are designed to deliver working software of the highest priority as fast as possible to the user, organizations

using these methods often lack grip on their software development budgets. Basically, organizations outsourcing their software development to agile teams provided by suppliers, went back to the time and material way of working of the nineties. This has resulted in a shift back to the customer of the risks associated with software development. The supplier gets paid no matter what, but if the productivity of the team is lower than required, the product that needs to contain the necessary functionality won't be ready when needed.

Unfortunately, even though all experts agree that this is an industry best practice, the Agile and DevOps communities have not adopted functional size measurement, as they don't perceive the use of function points on the team level. They mainly use story points, which is a subjective effort measure (not a size measure!) and use velocity based on story points as the main metric to show progress. Story point metrics work fine on the team level but are not useful for management as story points is not a defined standard and the number of story points can strongly vary per team. Thus, management often doesn't know the optimal team size and it's very hard to predict on which moment in time, which amount of functionality will be available and at which quality. Any software producing organization should be capable of demonstrating its productivity and benchmark this against industry peers. Story point metrics are not useful for this.

Especially in outsourcing contracts, organizations are struggling to put the right KPI's in contracts. The size of the software delivered (output) is an important measurement in many KPI metrics, such as Productivity (effort/output), Cost efficiency (cost/output) Speed/Velocity (output/month) and Quality (defects/output). When the output is not measured in a standardized, objective, verifiable, repeatable and therefore defensible way, metrics based on that output measure can't be used in contracts. This is one of the reasons many organizations nowadays contract agile teams based on rate cards (hour rates) per function/role/level, basically falling back to the risks associated with time/material contracts they faced in the eighties/nineties. They don't want this, but they don't see another way, while the solution is easy: use function points next to Story points in the agile teams. Story points for the agile team to commit to and to use in sprint planning, function points for long-term estimation, benchmarking, performance measurement and as a basis for contract KPI's.

The ISBSG data of completed projects, releases, and sprints, help organizations to regain their grip on application development, to measure and to improve productivity and to

(continued on next page)

(ISBSG Congratulates IFPUG, continued from page 23)

benchmark, no matter which development method used. This way, the risks in outsourcing are mitigated and significant cost savings are possible. Unfortunately, very few people understand this and know how to deal with this. ISBSG is now facing the challenge how to better align to Agile and DevOps teams, collecting relevant data in an automated way and to create meaningful products that give value to (management of) these teams.

ISBSG believes that function points are the solution to the lack of grip on agile team performance and that the main challenge is to explain to the world that function points are still relevant, maybe even more now than before!

Request for Volunteers

ISBSG is now looking for ways to engage with people in the Agile and DevOps community that acknowledge the idea that decisions should be made on data, not opinions. ISBSG needs to create ways to automate data collection from tools used in the agile community (e.g. Jira, Sonarqube, etc.) and it needs to create a vision on how to deal with size where functional size is not (yet) adopted. ISBSG wishes to help the industry to regain grip on software development, by providing relevant data and by creating valuable products. If you are interested, please send an e-mail to repositorymanagement@isbsg.org.

For more information about the ISBSG, please check www.isbsg.org.

IFPUG 30TH ANNIVERSARY – INTERNATIONAL YEAR OF SOFTWARE MEASUREMENT (IYSM) – RECALLING MY YEAR AS IFPUG PRESIDENT

Carol Dekkers, CFPS Fellow, PMP, AEC (Agile Expert Certified), P.Eng. (Canada)

President, Quality Plus Technologies, Inc.

IFPUG Director of Communications and Marketing

I remember with fondness my first IFPUG conference in 1992 in Phoenix, AZ, when IFPUG had emerged as the first truly global software measurement organization. I was so enthused about being a part of this exciting movement that I ran for and was elected to the IFPUG Board of Directors the following year, and proudly served as President in 1998-1999. (Side note, in November 2015, I was re-elected for the first time in 15 years to the IFPUG Board, where I now serve as IFPUG's Director of Communications and Marketing.)

1998 was a grand year as IFPUG accomplished :

- The 1st ISO standard on function points: ISO/IEC 14143-1: 1998 Functional Size Measurement – Part 1: Definition of Concepts;*
- >100 international volunteers “staffed” 10 IFPUG committees;*
- 2 conferences (spring and fall) with close to 400 participants each;*
- IFPUG had growing global affiliations (country-based) as well as several U.S. chapters;*
- Capers Jones published “Sizing up Software” in Scientific American;*
- The dominant IT concern was Y2K projects (and IFPUGgers weighed in variously about whether FPA could/could not help with estimating Y2K projects.)*

Since then, IFPUG and the software measurement world at-large have enjoyed success and setbacks, competition and collaboration, and, it seems that the more the world changes, the more it stays the same.

Today, as was true 30 years ago, managers want easy, quick, reliable and cheap “silver bullet” measures that solve the difficult problems of software estimating and productivity analysis, and function points (and SNAP points) can play a big part in the solution. We've seen Function Point usage and IFPUG membership grow the most in countries (and organizations) where FP are mandated as part of software development practices.

As we celebrate our 30th year of IFPUG (bylaws were incorporated in 1987), we still face an uphill battle to prove measurement ROI – but Function Points (and the emerging SNAP points) remain as relevant as ever. I remain involved (and open to your suggestions) to communicate the benefits and market FP and SNAP points to the world. There's no better time than today to join us as a fellow volunteer.

Committee Rosters

Certification Committee

- Gregory Allen, Pershing – **Chair**
- Mahesh Ananthakrishnan, Cognizant Technology Solutions
- Jim McCauley
- Teresa Cristina De Spagna Zenga Beraldo, BANCO BRADESCO S/A
- Francesco Gasparro, Capgemini Italia, SPA
- Manuel Buitrago, LEDA Consulting, S.L.
- Sheila Dennis, Premios
- Christine Green – Board Liaison

Communications and Marketing Committee

- Antonio Ferre Alberro, GFT IT Consulting – **Chair**
- Stephen Neuendorf, Premios – **Vice Chair**
- Paul Radford, Charismatek Software Metrics
- David Herron, Premios
- Justin Keswick, Bank of Montreal
- Tamara De San Teodoro, LEDA Consulting, S.L.
- Carol Dekkers – Board Liaison
- Linda Hughes, Accenture – Volunteer

Conference and Education Committee

- Filippo De Carli, GUFPI-ISMA Gruppo Utenti Function Point Italia – **Chair**
- Thiago Silva Da Conceicao, Synapsis
- Prof. Eduardo Alves De Oliveira, Servico Federal De Processamento De Dados (SERPRO)
- Antonio Ferre Alberro, GFT IT Consulting

- Dr. Luigi Buglione, Engineering Ingegneria Informatica SpA – Board Liaison
- Saurabh Saxena, Amdocs Ltd. – Volunteer
- Sushmitha Anatha, Accenture – Volunteer
- Alfonso González, LEDA mc – Volunteer

Functional Sizing Standards Committee

- Daniel Bradford French, Cobec Consulting – **Chair**
- Bonnie Brown, DXC Technology – **Vice Chair**
- Diana Baklizky, TI Metrics
- E. Jay Fischer, JRF Consulting
- Peter Thomas, Steria
- Adri Timp, Equens
- Roopali Thapar, IBM
- Tammy Preuss, AT&T
- Steve Keim, Premios
- Charles Wesolowski, SAIC
- Dácil Castelo, LEDA Consulting, S.L. – Board Liaison

International Membership Committee

- Saurabh Saxena, Amdocs Ltd. – **Chair**
- Iván Pinedo, Premios
- Bram Meyerson, Quantimetrix
- Anjali Mogre, Atos Origin International SAS
- Cao Ji, Beijing Suiji Tech
- Gianfranco Lanza, CSI Piemonte
- Marcio Silveira, DXC
- Pierre Almén, ImproveIT – Board Liaison

ISO Committee

- Steven Woodward, Cloud Perspectives – **Chair**
- Carol Dekkers, Quality Plus Technologies, Inc.
- Christine Green – Board Liaison

Non-Functional Sizing Standards Committee

- Talmon Ben-Cnaan, Amdocs – **Chair**
- Kathy Lamoureaux, DXC Technology – **Vice Chair**
- Dr. Charley Tichenor, Marymount University
- Francisco Julián Gómez, LEDA Consulting, S.L.
- Tomasz Marchel, Asseco Poland S.A.
- Roopali Thapar, IBM
- Jalaja Venkat, iGATE Global Solutions
- Saurabh Saxena, Amdocs Ltd.
- Sachin P. Thakur, Amdocs Ltd. – Volunteer
- Mauricio Aguiar, TI Metrics – Volunteer
- Dr. Luigi Buglione, Engineering Ingegneria Informatica SpA – Volunteer
- Dácil Castelo, LEDA Consulting, S.L. – Board Liaison

Congratulations to these NEW and Extended Certified Function Point Specialists!

Eder Artarxerxes Alves Mattos
Politec

Elisabetta Lattanzi
Green Aus S.P.A.

Sofia Quintão
Hitss Do Brasil Serviços
Tecnologicos Ltda

Achar Sreekanth
Accenture

Carlos Eduardo
Barros

William Matsuno

Bhuvaneshwari Senthilnathan

Brian Weisz
David Consulting Group

Caroline Domiciano
FATTO Consultoria E Sistemas

Alessandro Mei

Andrea Sisani
Green Aus S.P.A.

Ruth Wolf

Luciana Prata
Banco Central Do Brasil

Congratulations to these NEW Certified Function Point Practitioners!

Raffaella Casale
Green Aus S.P.A.

Ana Mara De Souza

Patrizia Mollicone
Green Aus S.P.A.

Júnio Marciel Rodrigues

Cristianne Couto Prado

Marco Di Tomaso
Green Aus S.P.A.

Rodolfo Pietropaoli
Green Aus S.P.A.

Márcia Maria Silva Souza
Maciel

Gesse Dafe
Radix Engenharia E Software

Giancarlo Furia
Green Aus S.P.A.

Massimo Poropat
Green Aus S.P.A.

Marco Todini
Green Aus S.P.A.

Deepa Dashetwar
Accenture

Michela Lucherini
Formit Servizi Spa

Paolo Rampazzo
Green Aus S.P.A.

Paolo Vertullo
Green Aus S.P.A.

Gabriel De Oliveira

Certification Matters!



Michael Harris, PREMIOS - USA

"Before I became the owner of DCG, my company was a client of DCG. DCG's analysis of our current state based on benchmarking a selection of our projects was fundamental to getting the investment approved for some significant process improvement. I would not have approved that investment without a level of certainty in the analysis that came from using IFPUG-certified function point analysts. Our clients insist on using certified function point analysts because we are often concerned with counting projects that are being delivered to fulfill a contract between our client and a third party. IFPUG certification means that both parties to the contract can be assured of the reliability of the counts. Indeed, IFPUG certification is written into those contracts!"



Congratulations CFPS Fellows throughout IFPUG's 30 Years

Anna Battistata
Informatica Trentina SPA

Massimo Beretta
SOGEI

Ray Boehm
Softcomptech, Inc.

Mary Dale
Q/P Management Group, Inc.

Carol Dekkers
Quality Plus Technologies Inc

E Jay Fischer
JRF Consulting

Loredana Frallicciardi
DDWAY S.R.L.

David Garmus
David Consulting Group

Roger Heller
Q/P Management Group, Inc.

Dan Horvath

Steve Keim
David Consulting Group

Makoto Kurashige
JFPUG-Japan Function Point
Users Group

Kriste Lawrence
DXC Technology

Lori Limbacher
PREMIOS

Debra Maschino
NASCO

Jim Mayes
CGI Group Inc

James McCauley

Mousa George Mitwasi
UnitedHealth Group IT

Stephen Neuendorf
Ford Motor Company

Bruce Paynter

Franco Perna
D.P.O. Srl

Bruce Rogora
Pershing LLC

Janet Russac
DXC Technology

Andrew Sanchez

Joanna Soles
WellPoint

Adri Timp
equensWorldline SE

Walter David Thompson

Steven Woodward
Cloud Perspectives

Daniele Zottarel
SOGEI

Congratulations to these NEW Certified SNAP Practitioners!

Ademir Aguilar
TI Metricas Ltda

Admar Andrade
Spread Sistemas e Automacao Ltda.

IN REMEMBRANCE

.....

On September 22, 2016, our friend David Thompson died suddenly of a stroke. David was a pillar of the IFPUG community, most recently serving as the Chair of the Communications and Marketing Committee. David engaged with everyone around him spreading thoughtfulness and joy.

Want to see your Company's Name here?
Advertise with IFPUG!

Advertising spots are available for the 2018 *MetricViews*. If you are interested in placing an ad please contact IFPUG headquarters at ifpug@ifpug.org



191 Clarksville Road
Princeton Junction, NJ 08550
USA