Soft Measures

A Total Metrics Newsletter Publication
Subscribe Free at: WWW.Totalmetrics.com

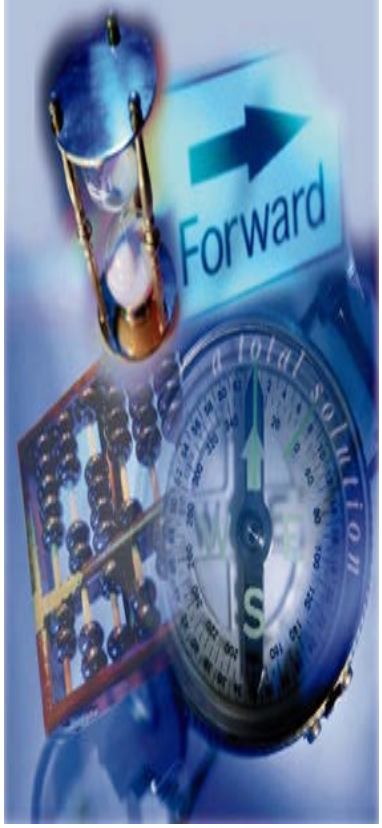
March, 2002

Volume 1, Issue 3

Special Interest Articles:

- Simplified project management reporting.
- Updates from IFPUG on certification credits
- Updates from ISBGS on their new data releases

Individual Highlights:



What's News in Metrics?

The Australian Software Metrics Association (ASMA) held its annual elections early March and has voted Pam Morris CEO of Total Metrics as its new Victorian President. Pam was one of the founding members of ASMA in 1990 and has been a continuing serving member on the ASMA executive and ISBSG committees. She is dedicated to ensuring that ASMA retains its place as an innovative leader in the international metrics arena.

Good news for organisations that require their IT processes to conform to ISO standards is that the voting for the acceptance of ***IFPUG 4.1*** and ***MarkII*** functional size methods has completed. ***Both*** methods have provisionally been approved as ISO standards.

The newer ***COSMIC-FFP functional size method*** is currently in its second last stage of balloting for ISO standardisation. Over the next few months we will keep you informed on the progress of all three functional sizing methods as they progress through the ISO process.

In this issue we have an update from the International Software Benchmarking Standards Group on products they will be delivering in 2002 to enable organisations to better understand the productivity of their software processes.

We commence Part 1 of our new series of articles on how to use function points beyond just input into estimation. These articles were developed by Total Metrics for the soon to be released IFPUG book on Practical Measurement.

Pam Morris CEO Total Metrics

Without objective data we are just another person with an opinion!

Historical data collected from actual software development projects is invaluable input for IT decision-making. Everyone wants information but hesitates at having to collect the data. However, despite these hurdles the International Software Benchmarking Standards Group, (ISBSG), has still managed to gather data on close to 1,400 software projects from more than 20 countries. But far more people want to devour information from the ISBSG analyses, than provide the data on which the analyses are based. One of the roles of the ISBSG is to provide a data collection "package". This package is effectively the data collection "standard" that presents questions to gather software project data that can be stored in the ISBSG (*.cont. Page 6)*

IFPUG Certification - An Exam free method!!

"IFPUG Certification Committee now issues Credits towards certification instead of re-sitting the exam"

Currently certified IFPUG CFPS practitioners can now earn certification credits that go towards their re-certification assessment. Earning 100 credits during the time that your certification is valid will enable you to continue to be certified without re-sitting the exam. Typically attendance at an IFPUG certified training course will earn you 10 certification credits.

Total Metrics currently delivers the IFPUG FP 331 training course that will earn you 10 credits.

September 2002 is the planned commencement date for the certification committee to start awarding credits for the following activities: teaching or authoring FP classes, attending or presenting FP related sessions, counting and validation activities etc.

The committee has made a special exemption to the commencement date so that attendees at the next IFPUG conference (April 2002) will be able to accumulate credits by attending specific accredited classes.

Renewal of your certification will cost the same irrespective of whether you sit the exam or accrue the 100 credits:

- $US150 for Members a
- $US300 for Non-members

There are some restrictions on using credits to re-certify for more details visit: www.ifpug.org

A simplified approach to Project Status Reporting

"Managing the quality and quantity of project reporting so that management can evaluate project status at a glance."

Jean-George Malcor is a busy man - as the Managing Director of the newly privatised ADI limited one of Australia's leading defence contractors with 2,500 personnel, he needs to know the 'health' of current software projects but does not have the time to read lengthy dissertations from project managers.

At the recent *6th Australian International Performance Management Symposium* in Canberra he presented an overview of the ADI project reporting system that enables him to evaluate the status of every project at glance, very quickly sorting the ones needing attention from the projects on track.

The ADI Project report is created by the project manager and the format is standardised to cover only three pages.

The first **Overview page** is narrative text under the following headings:

- Key achievements and or failures for the reporting period
- Key current issues
- Future potential issues with a forecast and action plan.

The second page is the **project's financial dashboard** includes charts of the profit and loss and the balance sheet. The graphs are annotated with a variance analysis providing an explanation of the reasons for the variance.

The final page is a **Risk analysis** with supporting data. It includes:

- Risk Log – describing the nature of the risk
- Risk Quantum – quantifying the size of the risk
- Mitigation Strategy – describing the consequences of the risk actually eventuating with a brief plan to avoid the risk or minimise the impact.

The report data is easily extracted from the ERP financial system and the Earned Value Management System.

The benefits of this approach are that the project managers are *routinely* forced to think *quantitatively* about their projects. Management can quickly review standard reports that display distilled, objective and useful data that is easy to understand.
Malcor warned that it was: "..critical for an organisation to have sufficient 'emotional maturity' to be able to use the project data for effective decision making"

Function Points – beyond project estimating – Part 1

Functional size measurement is the most commonly used method for sizing software. Most people use it to quantify project scope for input into estimates or to compare productivity.

More experienced users are finding they can use the size measure to assist them in a host of other management decisions.

This is the first of a multi-part article on practical uses of function point results.

Introduction

Industry experience has shown that an emphasis on project management and control offsets much of the risk associated with software projects. One of the major components of better management and control of both in-house development and a package implementation is ***measurement***.

This includes measurement of the:

- scope of the project (for example, software units to be delivered),
- performance indicators of efficiency and cost-effectiveness (cost per unit of software delivered,
- staff resources per unit of software delivered, elapsed time to deliver a unit of software), and
- quality indicators (such as number of defects found per unit of software delivered).

The outcome of a function point count provides the metric "unit of software delivered" and can assist in the management and control of software development, customization, or major enhancements from early project planning phases to the ongoing support of the application.

Knowing the software size facilitates the creation of more accurate estimates of project resources and delivery dates and aids in project tracking to monitor any unforeseen increases in scope. The measurement of the performance indicators enables benchmarking against other development teams and better estimating of future projects. These and other lesser-known ways in which FPA can assist IT to move toward best practice in the management of their software products and processes are discussed in this chapter.

Using measurement to support management decision-making is only successful if the information supporting these decisions is relevant, accurate, and timely. To ensure the quality of their measurement data, organizations need to implement a measurement process. The cost of implementing the activities, procedures, and standards to support the function point counting process depends on the size and structure of the organization and their measurement needs. These considerations are discussed in the last section "Cost of Implementing Function Point Analysis."

Managing Project Development

In the past the use of function point counts, within most organisations, has been restricted to input into estimating or benchmarking productivity rates. The examples in this chapter illustrate a wider range of uses where FPA can contribute to better management and control of

---

"Total Metrics new software scope management tool assists in the collection, analysis and reporting of software size to assist you in the type of decision making described in this article. Contact us for more information.

Admin@totalmetrics.com

CALL FOR PAPERS

ACOSM - 2002
Australian Software Metrics Conference – Melbourne Australia
November 19 –22.
Abstracts due:
May 31st 2002
Contact:
asmavic@ozonline.com.au

PMI PROJECT MANAGEMENT CONFERENCE
"The Value of Project Management"
Melbourne Australia
October 21-25 2002
MELBOURNE
The proposal
Abstracts due:
May 6th 2002
Contact:
seminars_world@pmimelb.org.au

…….continued

Function Points – beyond project estimating..... Continued

*"Function Point **Analysis** is just that – a quantitative means of **analysing** your software's functionality to provide objective input into decision making. Remember without measurement you are just another person with an opinion!"*

"International Software Benchmarking Standards Group Release 6 Report, April 2000, provides cost value for software projects in 1999.

Median costs to develop a function point = $US716,

Whereas the average cost was = $US849 per function point.

If you always do what you always did then you will always get what you have always got!

Start changing the way you work today!

the whole software production environment from project inception through to ongoing maintenance and support.

FPA Uses and Benefits in Project Planning

Function point counts can be performed as soon as the user has identified their functional requirements for a project. The count can provide input into the following early project decisions.

Project Scoping

A recommended approach for developing function point counts is to first functionally decompose the software into its elementary functional components (base functional components). This decomposition can be illustrated graphically as a functional hierarchy. This pictorial table of contents or map easily conveys the scope of the application to the user. Not only does it illustrate the *number* of functions delivered by each functional area but also the comparative *size* of each functional area, measured in function points.

Assessing Replacement Impact

If the software to be developed is to replace existing production applications, it is useful to assess if the business is going to be delivered more, less, or the same functionality. The replacement system's functionality can be mapped against the functionality of the existing system and a quantitative assessment of the difference measured in function points. Note that this comparison can be done only if the existing applications have already been sized in function points.

Assessing Replacement Cost

Multiplying the size of the application to be replaced by an estimate of the development dollar cost per function point, enables project sponsors to quickly estimate replacement costs. Industry-derived costs are available and provide a ballpark figure for the likely cost. Industry figures are particularly useful if the redevelopment is for a new software or hardware platform not previously experienced by the organization. Ideally, organizations should establish their own cost per function point metrics for their particular environment, based on project history.

Negotiating Scope

Initial project estimates often exceed the sponsor's planned delivery date and budgeted cost. A reduction in the scope of the functionality to be delivered may be necessary for delivery within predetermined time or budget constraints. The functional hierarchy provides the "sketch pad" for scope negotiation; that is, it enables the project manager and the user to work together to identify and flag functions that are *mandatory* for the first release of the application; *essential* but not mandatory, or *optional* and could be held over to a subsequent release.

The scope of the different scenarios can then be quickly determined by measuring the functional size of the scenarios. For example, the project size can be objectively measured to determine the size (and cost and duration) if all functions are implemented, only *mandatory* functions are implemented, or only *mandatory* and *essential* functions are implemented.

This allows the user to make more informed decisions on which functions to include in each release of the application, based on their relative priority and compared to what is possible given the time, cost, and resource constraints of the project.

Evaluating Requirements

Functionally sizing the requirements for the application quantifies the types of functionality delivered by an application. The function point count

….continued

Function Points – beyond project estimating..... Continued

assigns function points to each of the function types: External Inputs, Outputs and Inquiries, and Internal and External Files.

Industry figures available from the International Software Benchmarking Standards Groups (ISBSG) repository for projects measured with IFPUG function points indicates that complete applications tend to have consistent and predictable ratios of each of the function types. The profile of functionality delivered by each function type in a planned application can be compared to that of the typical profile from implemented applications to highlight areas where the specifications may be incomplete or anomalies may exist.

The pie chart in Figure 1 illustrates the function point count profile for a planned accounts receivable application compared to ISBSG data. The reporting functions (*outputs*) are lower than predicted by industry comparisons. Incomplete specification of reporting functions is a common phenomenon early in a project's life cycle and highlights the potential for substantial growth creep later in the project as the user identifies additional reporting needs.

The quantitative comparison shows that the reporting requirements were lower than expected by about half (14 percent compared to the expected 23 percent of the total function points). The project manager in this case verified with the user that the first release of the software would require all reporting requirements, and the user indicated that more reports were likely to be specified.

The project manager increased the original count to allow for the extra 9 percent and based early project estimates on the higher figure that was more likely to reflect the size of the delivered product. The function point measurement activity enabled the project manager to quantify the potential missing functionality and justify the higher, more realistic estimate.

Allocating Testing Resources

The functional hierarchy developed as part of the function point count during project development can assist the testing manager to identify high complexity functional areas which may need extra attention during the testing phase. Dividing the total function points for each functional area by the total number of functions allocated to that group of functions, enables the assessment of the relative complexity of each of the functional areas.

The effort to perform acceptance testing and the number of test cases required is related to the number and complexity of the user functions within a functional area. Quantifying the relative size of each functional area will enable the project manager to allocate appropriate testing staff and check relative number of test cases assigned.

Risk Assessment

Many organisations have large legacy software applications, that due to their age, are unable to be quickly enhanced to respond to the needs of their rapidly changing business environments. Over time these applications have been patched and expanded until they have grown to monstrous proportions. Frustrated by long delays in implementing changes, lack of support for their technical platform and expensive support costs, management will often decide to redevelop the entire application. For many organisations this strategy of rebuilding their super-large applications has proved to be a disaster resulting in cancellation of the project mid-development. Industry figures show that the risk of project failure rapidly increases with project size. Projects less than 1500 function points have a risk of failure of less than 20% in comparison with projects over 5000 function points which have a probability of cancellation close to 40% (Software Productivity Research (Capers Jones) - Softwares Chronic Crisis W. Gibbs Scientific American September 1994). This level of risk is unacceptable for most organisations.

Assessing planned projects for their delivered size in function points enables management to make informed decisions about the risk involved in developing large highly integrated applications or adopting a lower risk phased approach as described in the next newsletter June 2002.

Checking completeness of project requirements against ISBSG Release 6.0

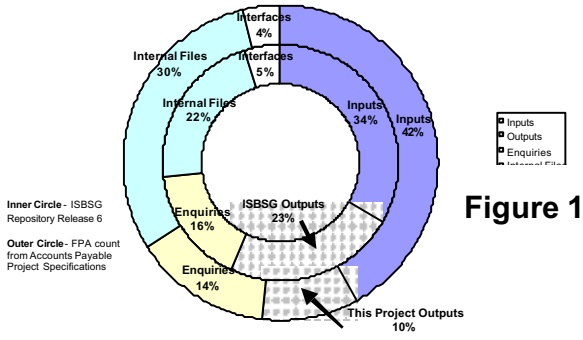


Figure 1

TOTAL METRICS

Suite 1
667 Burke Road
Camberwell
Victoria, 3124
Australia

PHONE:
61(0) 3 9882 7611
FAX:
61(0) 3 9882 7633
E-MAIL:
admin@totalmetrics
.com

We're on the Web!

Visit us and subscribe

www.totalmetrics.com

About Our Organization...

Total Metrics is a leading software measurement company. We assist our world wide clients to better manage and control their software application environment by measuring, monitoring and benchmarking their IT performance.

continued from page 1

publicly available *R*epository, then used for analysis. What can we discover from the data that is collected?

- ***What is an optimum team size for maximum productivity?***
- ***Does productivity improve with the use of CASE tools?***
- ***What development techniques are being used and what has been their impact?***
- ***Is outsourced software development more successful than software developed internally?***
- ***What accuracy of software estimation are companies experiencing***?

These are just a few of the many questions that software professionals are seeking answers to. The data collected by the ISBSG allows analysis to turn this data into information that provides the answers.

But the ISBSG data collection package is not restricted to the ISBSG itself. Any organisation can down load a copy of the package from the ISBSG web site FREE, (www.isbsg.org.au) and use it as the basis of their own project data collection. The ISBSG has just revised its collection package for New Developments & Enhancements. The new package will be available for down load in April. A data collection package for Maintenance and Support will be released later in the year.

Principles of good leadership

The following is an excerpt from a presentation on leadership by General Colin Powell Chairman (Ret), Joint Chiefs of Staff USA Military. Many of these concepts can be transferred to the corporate management environment to give us an insight into what makes an inspiring leader.

"You don't know what you can get away with until you try."

You know the expression; "it's easier to get forgiveness than permission." Well, it's true. Good leaders don't wait for official blessing to try things out. They're prudent, not reckless. But they also realize a fact of life in most organizations: if you ask enough people for permission, you'll inevitably come up against someone who believes his job is to say "no." So the moral is, don't ask. Less effective middle managers endorsed the sentiment, "If I haven't explicitly been told 'yes,' I can't do it," whereas the good ones believed, "If I haven't explicitly been told 'no,' I can." There's a world of difference between these two points of view.

"Never let your ego get so close to your position that when your position goes, your ego goes with it."

Too often, people who cling to familiar turfs and job descriptions stifle change. One reason that even large organizations shrivel is that managers won't challenge old, comfortable ways of doing things. But real leaders understand that, nowadays, every one of our jobs is becoming obsolete. The proper response is to obsolete our activities before someone else does. Effective leaders create a climate where people's worth is determined by their willingness to learn new skills and grab new responsibilities, thus perpetually reinventing their jobs. ***The most important question in performance evaluation becomes not, "How well did you perform your job since the last time we met?" but, "How much did you change it?"***

Thoughts for the Month

"A fine is a tax for doing *wrong*. A tax is a fine for doing *well."*

"The sooner you fall behind, the more time you'll have to catch up."

"To steal ideas from one person is plagiarism; to steal from many is research."

"A conclusion is the place where you arrived when you got tired of thinking."

"The bitterness of poor quality remains long after the sweetness of low price is forgotten."