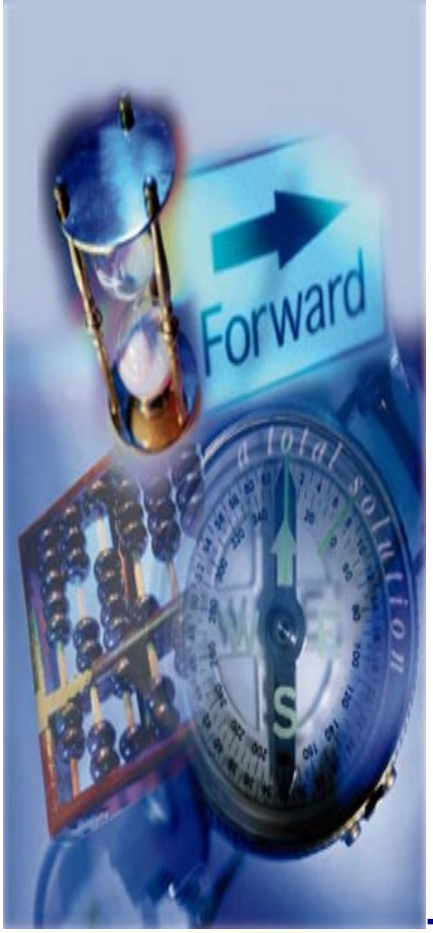
Soft Measures

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

June 2003
Volume 1, Issue 5

Special Interest Articles:

Individual Highlights:



What's News in Metrics?

Our most significant news for this month is the availability of Release 8 of the International Software Benchmarking Groups Data Repository data listing project data from over 2000 projects. Visit : WWW.ISBSG.org for details

Total Metrics is pleased to announce their appointment to the Victorian Government's Eservices Panel. We are the only specialist metrics company to be approved by the state government as a preferred supplier of measurement consultancy and training in their key services. Category See www.mmv.vic.gov.au.

We conclude our 3 part series of articles on how to use function points beyond project estimation. The final part looks at how FPA can assist in the evaluation of supplier support contracts, comparison of functionality offered by software packages and valuing your software assets.

Of the $382 billion spent on software development in the United States last year, it is estimated that $82 billion was wasted through bad planning and mismanagement. Our special interest article this month looks at the role that metrics can play in improved project governance.

Pam Morris

Editor

Metrics based Project Governance

In our last issue we quoted Ed Yourdon identifying the key reasons why projects failed. These included inadequate estimates, failing to measure requirements churn and lack of clear acceptance criteria for when a project is finished. The *2003 Chaos Standish Report* found that only 34% of projects finished successfully and 15% were cancelled altogether.

These observations and figures are supported by our experience at Total Metrics where we find the most successful projects are those that implement measurement at feasibility stage, develop accurate auditable estimates and manage scope and changing requirements.

Our local state government has recommended that all outsourced government sponsored software projects are to have an independent metrics expert to act as a *Scope Manager* in a governance role to try and optimise the chance of the project's success. See www.mmv.vic.gov.au/southernSCOPE).

The role of the *Scope Manager* starts at **Feasibility Stage** where

…. continued

Metrics based Project Governance - continued

"The International Software Benchmarking Standards Group (ISBSG) provides industry data from over 2000 recently completed projects worldwide. The data can be used to externally verify project estimates and vendor prices and schedules.

International Software Benchmarking Standards Group

ISBSG

For information on how to access the ISBSG data, visit : www.ISBSG.org.au.

At Total Metrics we say "if we cannot count it then the developers cannot build it!". Specifications for functions, that cannot be accurately measured and assessed for complexity, are flagged as part of the count and returned to the authors for further review and refinement. The Baseline count is not approved by the Scope Manager until Specifications are complete.

they estimate the functional size of the project and use historical industry data (see ISBSG) to provide ballpark estimates.

These estimates, based on actual past projects, provide an independent reality check against the project sponsor's estimates, which are often too optimistic and fail to take into account unexpected but commonly occurring events.

If the project is approved then at the **Requirements Specification Stage** the *Scope Manager* reviews the requirements for completeness, clarity and internal consistency. The review is done concurrently with measuring the functional size. At Total Metrics we say "*if we cannot count it then the developers cannot build it!".* Specifications for functions, that cannot be accurately measured and assessed for complexity, are flagged as part of the count and returned to the authors for further review and refinement.

The Baseline count is not approved by the Scope Manager until Specifications are approved as complete.

The outcome from a function point count is a complete list of catalogued and classified functions (processes and data).

The list may be further analysed with the user to identify those functions that are critical to the required software product, or optional or just 'nice to have'. The list of functions, their priority along with the functional size is included in the **Request for Tender** that is sent to prospective software vendors. It provides an auditable, traceable list of functionality that can be later checked against the functionality actually delivered.

At **Tender Evaluation stage** the *Scope Manager* compares the vendors planned solution against the functional list identifying the proportion of required functionality that is proposed to be delivered as part of their solution. If different delivery mechanisms are also planned then the *Scope Manager* will flag the functionality and quantify how much functionality will be delivered by different means. Ie, 200 function points by the package, 50 function points customised package functionality and 100 function points purpose built. Each delivery mechanism will have different productivity rates applied when developing estimates. Often the vendors quote a fixed price of '*dollars per function point*' for the project. The *Scope Manager* assesses the reality of the vendor's quotation by comparing the vendor's price and their planned schedule to the estimates derived from the productivity rates and industry data.

The selected vendor will then assist with the development of the **Functional Specification**. This is often done at a fixed price or it may be part of their original $/fp quotation.

The *Scope Manager* reviews the quality and completeness of the functional specification and cross-references it to the original requirements list, to ensure all the user requirements are included.

If the scope of the project has changed then the estimates need to be revised. If the new estimates exceed the budget and schedule, then the *Scope Manager* will work with the User and Vendor to reduce the scope to fit within the constraints.

The *Scope Manager* only approves and recommends that the

….continued Page 6

Function Points – beyond project estimating – Part 3

Functional Sizing enables software managers to gain a quantitative handle on size of their software applications. They can use this objective measure of functionality to evaluate supplier support contracts and compare functionality offered by software packages.

The conclusion of our 3 part series on the uses and benefits of function points explains how.

"Functional size enables software assets to be objectively valued as a depreciable 'asset', thus enabling your costs to be amortised over 5 years and the negative impact on your company's profit to be more evenly spread".

Introduction

This is the conclusion of our three part series published over the last three issues of *Soft Measures*. Each part has focussed on a different part of the software development life cycle with respect to the different:

- benefits that can be achieved by an organisation that collects function point measures.
- uses for function point analysis beyond simply measuring productivity and improving project estimates.

In this issue we look at the ways function points can assist in better management and control of software development projects after software implementation.

The concepts explained are applicable for all functional size measurement (FSM) methods.

Software Asset Valuation

Function Point Analysis is being used increasingly by organizations to support the valuation of their software assets. In the past, software was considered an expense rather than a capital asset and was not included in an organisation's asset register.

The most commonly used software valuation method is based on the deprival method, which values the software based on what it would cost to replace in today's technical environment rather than what it cost originally to build. The industry build rate (dollar cost per function point) is determined, and the total replacement value is calculated based on the current functional size of the application.

Because FSM provides a means of reliably measuring software, many organisations, particularly in Government, have implemented accrual budgeting and accounting in their business units. Under this strategy, all assets must be valued based on deprival value and brought to account, thus ensuring better accountability of the organisation's spending.

Funding via budget allocation is based on assets listed in the financial accounts and their annual depreciation. In the past, the purchase price of the software was recorded as an expense within an accounting year. These more recent accounting practices mean that it can now be valued as an asset and depreciated.

This is particularly attractive to publicly listed organizations who found that using this accrual accounting method of measuring software as an asset rather than an expense lets them amortise the depreciation over five years rather than artificially decreasing the current year's profit by the total cost of the software. This strategy has a dramatic

……..continued

Function Points – beyond project estimating – Part 3..... Continued

"Function Point Analysis provides an objective quantitative means of establishing the scope of the applications portfolio in due diligence stage of outsourcing negotiations!"

"*Many outsourcing agreements establish penalty and bonuses to be paid depending on whether the supplier achieves target productivity rates. It is important that the client and the supplier agree up-front on what are included in function point generating activities and what are not!*

"*Functional size measurement provides a measure of the amount of software supported in an outsourcing contract. This enables you to compare the $ cost per function point you are paying your suppler to standard industry rates. It enables the client to negotiate price from an objective number rather than relying on the supplier estimates."*

effect on their share price because software listed as a capital asset contributes to the overall worth of the company, and the total cost of that asset has a reduced impact on the current year's reported profit.

Outsourcing Software Production and Support

In outsourcing contracts, functional size enables suppliers to measure the cost of a unit of output from the IT process to the business and to negotiate outcomes with their client. Specifically, these output-based metrics based on function point analysis have enabled suppliers to:

- Quantitatively and objectively differentiate themselves from their competitors.
- Quantify the extent of annual improvement and achievement of contractual targets.
- Negotiate price variations with clients based on an agreed metric.
- Measure financial performance of the contract based on unit cost of output
- Be in a stronger bargaining position at contract renewal, supported by an established set of metrics.

Conversely, these output-based metrics based on function point analysis have enabled clients to:

- Objectively assess supplier performance based on performance outputs delivered rather than concentrating on inputs consumed.
- Establish quantitative performance targets and implement supplier penalties and bonuses based on achievement of these targets.
- Measure the difference between internal IT costs compared to the cost of outsourcing based on similar output.
- Quantitatively compare competing suppliers at the contract tender evaluation stage.

Most of the international outsourcing companies use function point metrics as part of their client service-level agreements. Although this method of contract management is relatively new, its proponents verify its usefulness. In our experience, once an outsourcing contract has been based on function point metrics, subsequent contract renewals expand on their use.
Metrics initiatives have a high cost and need substantial investment, which is often overlooked at contract price negotiation. Typically, both the supplier and the client incur costs. However, given the size of the penalties and bonuses associated with these contracts, the advantage of the investment is obvious.

Customizing Packaged Software.

For selected MIS applications, implementing a packaged off-the-shelf solution is both cost effective and time efficient for delivering necessary functionality to a business.
All the benefits and uses of function point analysis described for in-house development projects in the previous section can also apply to projects that tailor a vendor-supplied package to an organization's specific business needs.
Experience shows that function point counting of packages is not always as straightforward as sizing software developed in-house, for the following reasons:

- Only the physical and technical functions are visible to the counter. The logical user view is often masked by the physical implementation of the original logical user requirements.
- In most cases the functional requirements, functional specifications, and logical design documentation are not delivered with the software. The counter may have to rely on the user manual or

….continued

Function Points – beyond project estimating – Part 3- Continued

online help to assist in interpreting the user view.

- Modelling of the logical business transactions often requires the function point counter to work with the client to identify the logical transactions. They do this by investigating the user's functional requirements and interpreting the logical transactions from the package's physical implementation.

In most cases, the names of the logical files accessed by the application's transactions are not supplied by the package vendor.
The function point counter needs to develop the data model by analyzing the data items processed by the application.
However, with sufficient care, you can obtain a reasonably accurate function point count of packaged applications.

Estimating Package Implementations

The project estimates for a package solution need to be refined for each implementation, depending on the percentage of project functionality in the following categories:

- Native to the package and implemented without change.
- Within the package and needing to be customized for this installation.
- Contained within the organization's existing applications and needing to be converted to adapt to the constraints of the package to be built as new functions, in addition to the package functions to be built to as new functions, to enable interfacing to other in-house applications not to be delivered in this release.

The productivity rates for each of these development activities (to implement, customise, enhance, or build) are usually different. Assigning an appropriate productivity factor becomes even more complex when the package provides utilities that enable quick delivery based on changes to rule-based table. Change requests that can be implemented by changing values in rule-based tables are very efficient compared to similar user change requests requiring source code modification. These activities should be identified and effort collected against them accordingly to determine productivity rates for the different activity types.
Functions can be flagged for their development activity type and their relative contributions to the functional size calculated. This enables fine-tuning of the project estimates.

Another area of concern when developing estimates for package integration is the need to determine the extent to which the application module needs to interface with existing functionality. The function point count measures the external files accessed by transactions within this application. A high percentage of interface files (>10 percent) suggests a high degree of coupling between this application and existing applications. A high degree of interfacing tends to have a significantly negative impact on productivity rates and should be considered when you are developing estimates.

Alternatively, it may highlight a need to review the functionality to be delivered by this release.

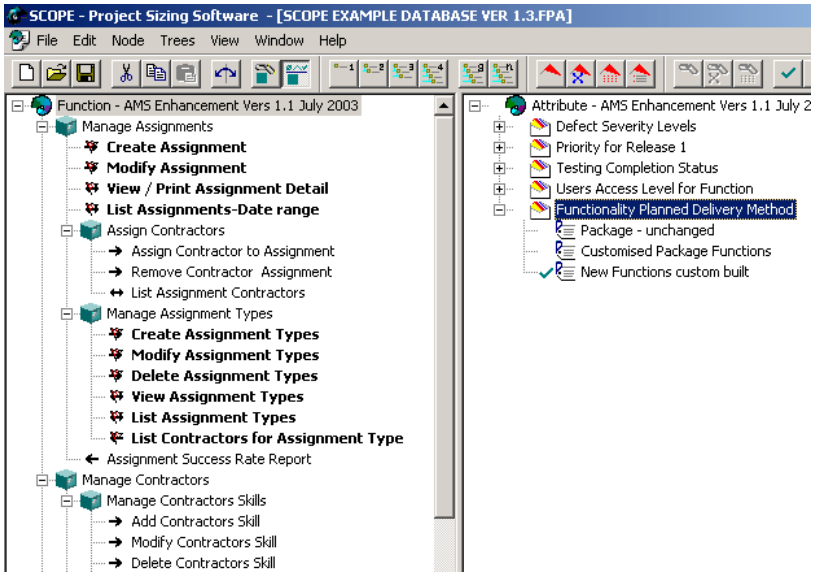


Figure 1

Identify which functions are delivered by each delivery mechanism. Apply different productivity rates to each type of delivery for more accurate estimates.

Next Issue our feature article describes strategies to count many applications within a constrained time and budget.

Metrics based Project Governance continued

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 FREE

www.totalmetrics.com

About Our Organization...

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

Thoughts for the Month

" Most of us go to our grave with our music still inside of us."

" Some mistakes are too much fun to only make once."

" Birthdays are good for you; the more you have, the longer you live."

" A truly happy person is one who can enjoy the scenery on a detour."

project proceed when all requirements are clearly stated and the functional specification is internally consistent.

The baseline functional size is established and confirmed by all parties. The change control model and the pricing model ($/fp) for changes in functionality is agreed.

During the **Build Stage** the *Scope Manager* assesses any proposed changes for their scope of impact on the baseline functionality.

They provide the client with the size of the change and the calculated price based on function points and the pre-agreed penalty rate in $/fp for the stage of the project. The *Scope Manager* also estimates the predicted impact on the schedule. The client can decide to confirm or withdraw the change.

The *Scope Manager* assists the project manager in collecting measures on which functions in the list have achieved which project milestone. 'Earned Value' type **Project Status Reports** identify the percentage of functionality delivered (%fps), compared to the percentage predicted. This quantitative, objective reporting ensures early warning of project slippage.

Once the project is **implemented** the *Scope Manager* assesses the actual function points implemented and assists in final price negotiations. The *Scope Manager* can assist in collection and submission of project data for the International Software Benchmarking Standards Group's (ISBSG) Database.

ISBSGs produce a free report that assesses and compares the project productivity and quality to other similar industry projects.

Summary

The *Scope Manager* acts in a role that combines the activities of quantity surveyor and an independent auditor. They bring their extensive metrics and analysis skills to the project to evaluate and independently quantify the product. They produce an objective quantified, traceable and auditable list of requirements that the client requires to be delivered. This rigour ensures that the client only accepts and pays for what was requested.

Ideally the *Scope Manager* reports to the project board rather than to the project team, since often the project team have a vested interest in the project proceeding and tend to disregard early warning of problems or risks. The project board can accept or reject the *Scope Manager's* recommendations but do so at their own peril!

ContactTotal Metrics for more information about our experienced Scope Managers

ACOSM – 2003 METRICS CONFERENCE

Australian Software Metrics Association (ASMA) Annual Conference Sydney Australia September 3rd – 5tth 2003

- Workshops
- Presentations

Book now
http://metric.cse.unsw.edu.au/Metrics2003

IFPUG – 2003 METRICS CONFERENCE

International Function Point Users Group (IFPUG) Annual Conference Scottsdale Arizona USA September 13th – 19th 2003

- Workshops
- Presentations
- CFPS Exam

Book now
http://www.ifpug.org/conferences/annual.htm