



# Metrics Based Project Governance

Version 1.0 - November 2004

Pam Morris (BSc.Grad Dip Comp.Dip Ed, CFPS, CSMS (Level 3))

Total Metrics (Australia)

See [www.Totalmetrics.com](http://www.Totalmetrics.com) for a copy of this paper and other resources

Email : [Pam.Morris@Totalmetrics.com](mailto:Pam.Morris@Totalmetrics.com)

## **Abstract:**

*This paper describes a rigorous approach to software development project control by introducing functional size measurement at the planning stage and objectively quantifying the status and scope of the project and its deliverables throughout its lifecycle. The Scope Manager's role is both that of 'quantity surveyor' and project auditor. The paper defines this role and describes the process and the benefits of the metrics they bring to a project. The paper discusses how organisations typically do not harness the potential of the metrics personnel in their organisation to pro-actively assist project teams manage risk. The engagement of independent Scope Managers on a project assists in assuring successful projects or providing early warning of projects in trouble.*

## **Keywords**

*Functional size, project governance, software measurement, project estimation, requirements management.*

## **About the Author**

*Pam is the Managing Director of Total Metrics and past president of the Australian Software Metrics Association (ASMA) where she currently holds a position on their Executive and the Benchmarking Database Special Interest groups. In 2007 she was elected as Vice President of the International Software Benchmarking Standards Group (ISBSG) and is on the ISO/IEC Study group for the NWI Benchmarking Standard. She represents Standards Australia as the international project editor of the ISO standard 14143-1 and 2 for Functional Size Measurement. She was the international convenor of ISO/IEC/WG12 group developing FSM standards from 1997 to 2004. She plays an active role internationally in the development of measurement standards and was a member of the International Function Point User Group (IFPUG) Counting Practices Committee in the USA from 1993 to 2000 and is a reviewer of the IFPUG documents. She is a member of the COSMIC-FFP Core Group who is responsible for in the development of the COSMIC-FFP FSM method. She has been an IFPUG Certified Function Point Specialist (CFPS) since 1994, and a COSMIC Certified Practitioner and a Certified Software Measurement Specialist (CSMS Level 3) since 2006. In 2006 Pam was awarded the Australian ITP Lifetime Achievement Award for her services to the IT Industry. In 2007 Pam was invited to be an international expert partner of the Chinese Software Benchmarking Standards Group. Pam is a regular guest speaker on the topic of software metrics numerous international conferences in the USA, China, Japan, India, South Korea, New Zealand, Germany, South Africa, Spain, Italy and the UK.*

## 1 Background

Our experience as Software Measurement consultants has taught us that knowledge gained in the *act* of measurement can be as valuable to project managers, as their use of our measurement *results*.

For example, as part of our measurement role we are often asked to functionally size a project early in its lifecycle for input into estimation models. In order to perform the functional size measurement we need to dissect the functional requirements for the project, model them and individually catalogue and quantify each base functional component. In doing so we make observations about the status of the project and the quality of both the requirements and the specifications and based on our experience with similar projects, we are able to make predictions on the likelihood of the project's success.

With time, as our predictions have become reality, we have begun to recognise that our observations, if reported, could have provided valuable input into the overall management of the project and if heeded could have prevented numerous project failures. The *act* of measuring provides quantitative assessments of the quality and status of a project that is rarely seen by others and the measures we take give insights that can provide real benefits to the project team beyond just monitoring their productivity.

This realisation has caused us to change our approach and to focus on using the *role* of measuring to provide ongoing advice to the project whilst it is progressing. We use the measurements to quantify our observations so our recommendations are now a critical part of the project decision making. Metrics experts observe and measure without any vested interest and as such provide unbiased and independent assessment of the project risk, quality and status. The measurement results support these observations.

As metrics consultants we had been confusing the *measure* with the *goal*. Key objectives should not be the measurement results but *successful* software development projects. Delivery on time and on budget with the right functionality is the true measure of the worth of 'metrics' to a project. To make measurement an integral part of the software development process we first have to make it relevant to the project's success.

This change of emphasis has meant that we are now perceived by our clients as providing project governance through the act of measurement and called "Scope Managers" not "Metrics Consultants". We have shifted the perception of the project team from seeing us as just another project overhead, to being someone that can assist project teams to better manage and control their risk and optimise their chances of success.

## 2 The Role of the Scope Manager in Project Governance

The role of Project Governance is to balance the risk of the organisation's investment against the opportunities and benefits that the outcomes will provide the business. It addresses the

risks to ensure that the software provides value to the organisation and that the risks are properly mitigated.

For IT projects it is about providing the client with the capability to monitor the project status and control the risk of the project not delivering the business value they require within the time and budget available. Project governance is about understanding the business opportunities that the project can deliver, but also appreciating the consequences of failure and putting in place strategies to minimise the risk and optimise the investment so that business goals are achieved.

The Scope Manager provides metrics based project governance. We have found this approach to be very successful in objectively quantifying key project attributes to enable informed decision making with respect to project estimates and project risk.

The Scope Manager is typically a metrics specialist who has excellent skills in business analysis, project estimation and functional size measurement. They need to be independent of the project team and not be connected to either the IT developers or the business client. They

have to be able to report the status of the project objectively without bias, to a management level that has the authority to proceed, change direction or cancel the project.

The Scope Manager provides a key role in that their reports can provide early warning of project failure. In the past it was often not until major project milestones were missed that senior management had an indication that their investment was at risk.

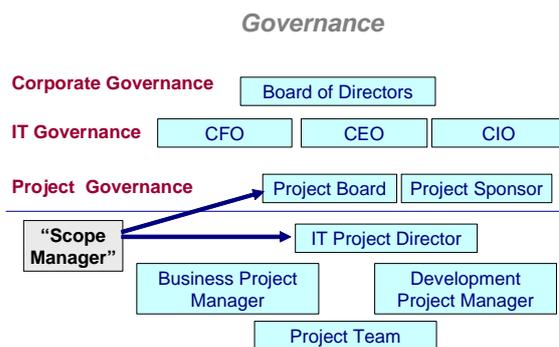


Figure 1 Scope Manager Reporting Level

The following sections describe the role of the Scope Manager over the life cycle of

the project and how they use metrics as a basis for their recommendations.

### 3 Role of Scope Manager During Project Life Cycle

#### 3.1 Business Planning / Feasibility

The Scope manager can be involved in the project as early as the business case stage where they assess the high-level business requirements to provide an estimated functional size of the proposed project. The functional size combined with a productivity rate for the planned development environment can be used to establish a ballpark range of predicted project effort, cost and likely duration.

If the organisation has its own internal productivity data then this can be used for the estimates. Alternatively, industry data for productivity rates are available from the International Software Benchmarking Standards Group (ISBSG) [1]. The ISBSG's data

provides industry productivity data for a wide range of development platforms, languages and environments.

The project estimates can be used as a ‘reality check’ against the planned budget and required delivery dates. If delivery time is constrained, then the Scope Manager can use ISBSG’s regression equations to demonstrate the trade-offs between compressing the schedule and the cost of adding more people on the project. For example, doubling the speed of project delivery requires up to four times the number of people [5]. Large teams have a significant negative impact on an individual’s productivity rate and consequently an overall increased cost of the project to deliver the same product.

If the estimated cost and duration exceeds the planned budget or schedule then the functionality may need to be reduced. Other governance processes need to be in place to ensure that the reduced functionality still delivers the planned business benefits.

Project risk of failure increases exponentially with project size. Early quantification of the size of the proposed software product enables evaluation of potential risk. The Scope Manager provides quantitative input for the business to make objective decisions as to the development strategy to minimise risk, whether to proceed to the next step of building a requirements specification or to cancel the project.

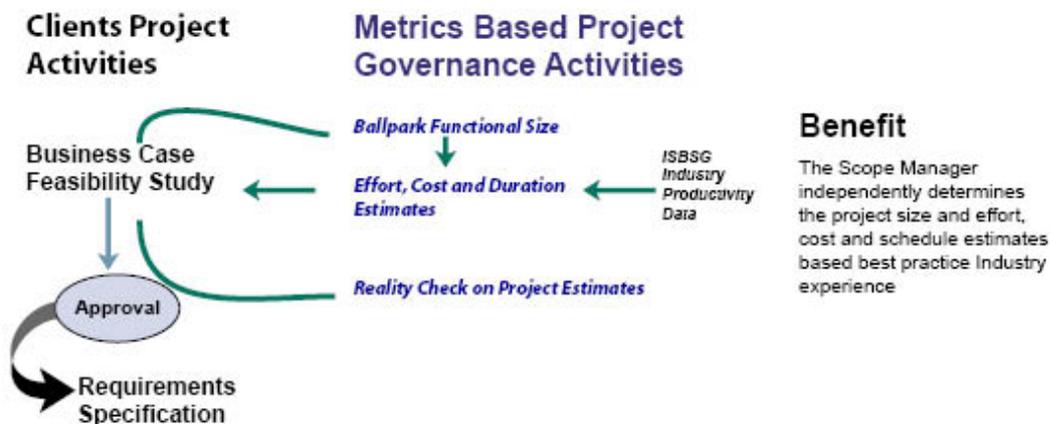


Figure 2 Scope Managers Activities at Feasibility Stage

### 3.2 Requirements Specification Stage

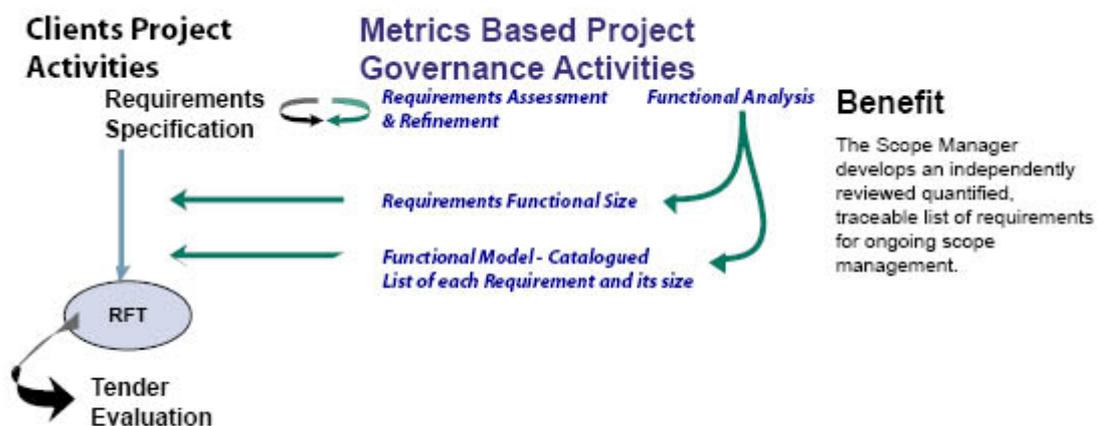
As part of the functional sizing process, the User’s Requirements need to be decomposed into individual functions within a functional model. Each function (process and data group) is identified, catalogued and sized. The cataloguing and modelling process often highlights gaps in the Requirements Specification ie, where functions have failed to be specified, or have been specified inadequately, inconsistently or ambiguously. The Scope Manager is in a unique independent position to view the project how the external developers may see it. The Scope Manager's experience with sizing functional specifications enables them to identify areas that

may have been overlooked by the project team and provide objective feedback on the quality of the specification. For example, they can mark up the functional model for functionality that has been explicitly specified or only implicitly specified and quantify the percentage of each. The functional size is still only an estimated 'range' as the complexity of many functions can often not be evaluated at this stage; it is usually anticipated that the project will grow further. The Users may also prioritise their Requirements as those that they consider to be *Core* functionality and mandatory to be delivered versus those that they consider to be *extended* or for future consideration. The Scope Manager can determine the size and estimate of each alternative.

High-level project resource estimates are revised based on the selected platform and the predicted size range. Once the project team have updated the specification to fix ambiguities, inconsistencies and missing functionality, the refined Requirements Specification is ready to be used as the basis for input into the Functional Specification. In an outsourcing situation the Requirements Specification would be provided as part of the Request for Tender (RFT). The functional sizing model along with its list of individual identified quantified functions and their associated priority for delivery is distributed as part of the RFT. This becomes the baseline Requirements document with which the business can evaluate whether the completed project has delivered their required functionality.

If the method of quotation by the suppliers is to be via a '*fixed \$ price per function point*' as identified within the SouthernSCOPE [5] methodology, then the tendering suppliers need a clear indication of which of the Users Requirements would be considered to be included or excluded from the fixed price.

The Scope Manager identifies which of the User's Requirements will consume effort (and therefore costs) that are proportional to the overall functional size and which will not, and thus be excluded from the fixed price. For example, documentation of Project Deliverables is proportional to functional size and would be included within the fixed price per function point. In comparison, research and acquisition of hardware is not, and should be quoted separately.



**Figure 3** Scope Managers Activities at Requirements Specification Stage

### 3.3 Supplier Selection Stage

The early ballpark estimates of projected effort, duration and cost based on functional size enable the client to objectively evaluate the ‘reasonableness’ of the supplier's proposed quotation and solution. This mitigates the risk of selecting the supplier based on the lowest price and promised fastest delivery who would potentially have the greatest risk of failing to deliver the project.

The Scope Manager uses the functional size model to quantify the ‘fit’ of each supplier’s proposed solution to the original requirements enabling full objective evaluation of the supplier’s solution by the quantification of the proportion of extra functionality, functionality omitted, functionality delivered by a package or functionality that needs to be customised or built.

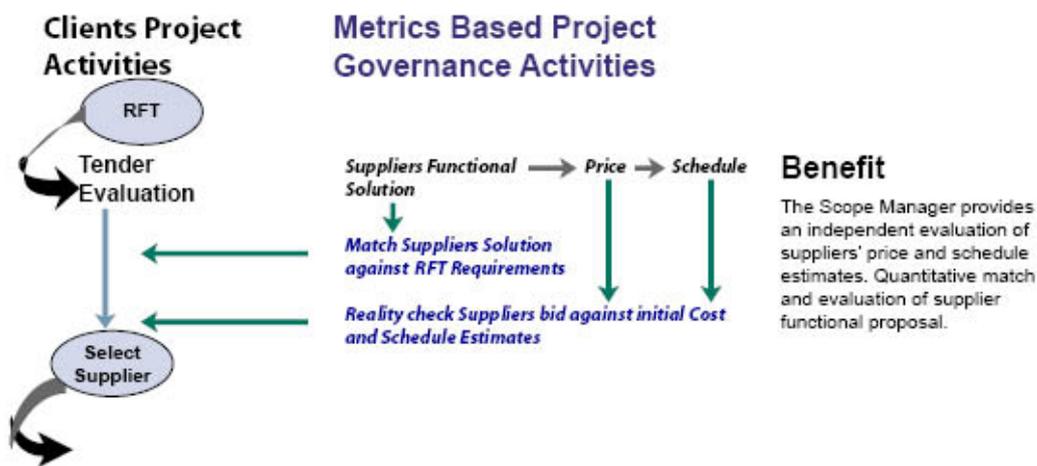


Figure 4 Scope Managers Activities at Supplier Selection Stage

### 3.4 Functional Specification Stage

The Scope Manager revises the functional size based on the Functional Specification and quantitatively maps the functional requirements to the original RFT Requirement’s Specification to provide a percentage match of the RFT to the proposed solution. Any omissions, ambiguities or inconsistencies in the Functional Specification are highlighted for revision before proceeding with the build. If at this stage the functional size indicates that the project will cost more, or be delivered later than planned, then non-core functionality is selectively removed from the project until the project size indicates that it can be delivered within the allocated budget and delivery dates.

If the project-charging model is based on *dollars per function point delivered*, the Scope Manager will work with the client and supplier to finalise the price variation model for changes that are approved during the remaining development. Ie, typically penalties are paid for any function points added, modified or deleted from this stage forwards. The dollar amount charged is usually scaled to increase as the project progresses. The outcome from the functional sizing and mapping exercise is a traceable, auditable, quantified list of agreed functional requirements to act as a baseline for ongoing scope management.

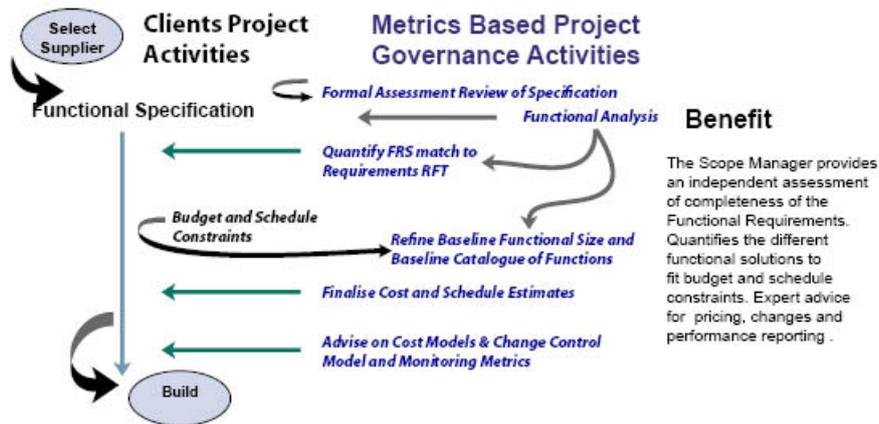


Figure 5 Scope Managers Activities at Functional Specification Stage

### 3.5 Changes introduced during the Project Build to Implementation

The Scope Manager is tasked with the quantification of Client Change Requests based on functional size of impact. This is used as a basis for pricing negotiations, enabling the client to assess the price of Change Requests prior to submission to the supplier and know they are being fairly charged for their required changes.

The Scope Manager uses the size of the change to establish the revised project scope as a means of evaluating the supplier’s revised project delivery date.

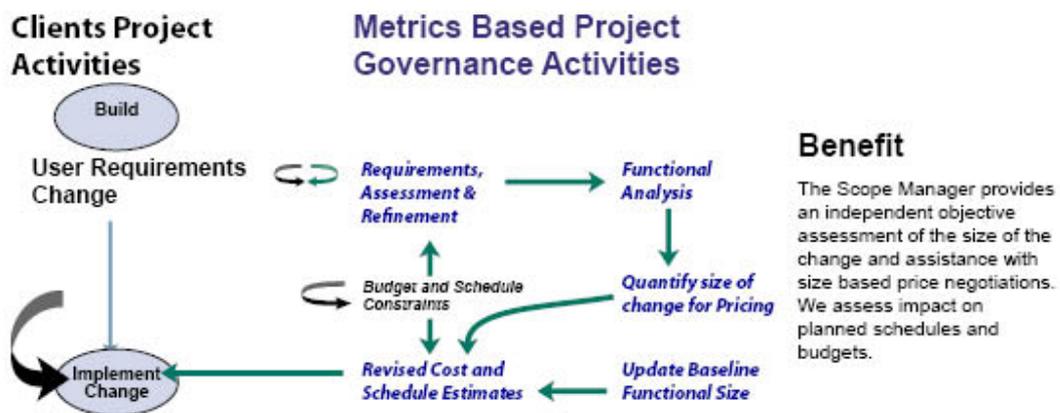
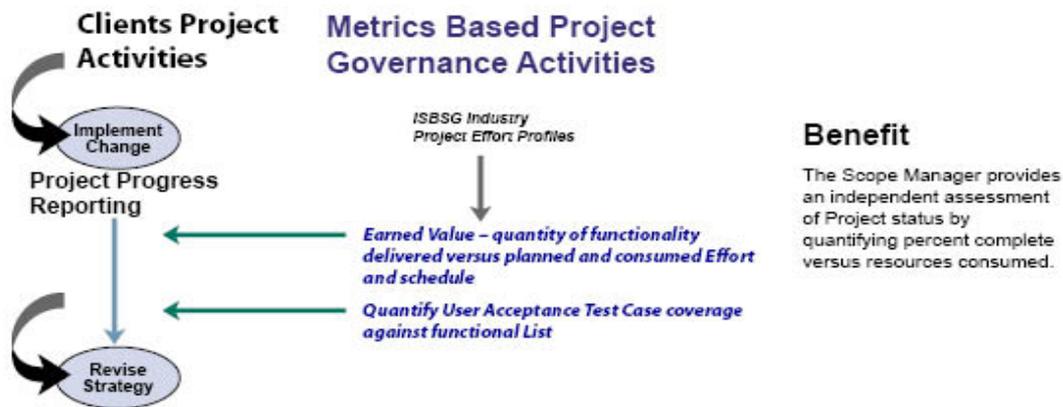


Figure 6 Scope Managers Activities for Change Control

### 3.6 Ongoing Project Monitoring

The Functional Size Model provides input into the quantitative monitoring of project status using an ‘earned’ value type of reporting [2], [3]. Ie, the Scope Manager provides independent project status reports based on the amount of functionality delivered, versus functionality planned to be delivered, within each reporting period. This is an ‘output based’ metric for project reporting that is more meaningful to the business client rather than an input based metrics of budget or effort consumed. Ie, status reporting is based on the amount of product delivered (function points) to each stage of completeness. This contrasts with traditional approaches of monitoring status based on resources and schedule consumed.

The status report provides the client with detailed objective independent assessment about which functions of their software have been developed to what stage. The increased visibility of project status gives early warning of project slippage.



**Figure 7** Scope Management Activities for Project Status Monitoring

### 3.7 Project Implementation

On project completion the Scope Manager quantifies and maps the functionality implemented versus functionality contracted to be delivered, for input into final payment negotiations. This enables the client to verify, against the traceable list of requirements, which functions have been satisfactorily delivered. The quantification of the delivered functionality enables objective discussions on payments due.

The Scope Manager provides advice on the project metrics to collect, analyse and report, and ensures that they are consistent with the organisations internal standards or those of ISBSGs. The Scope Manager can assist with the submission of the project data to the ISBSG repository and provide an independent assessment of the developers productivity and product quality.



**Figure 8** Scope Management Activities at Project Implementation

## 4 Conclusion

The Scope Manager focuses on the effective management and control of the project and uses their metrics skills to provide objective evidence of their observations, shifting the focus from measurement to project governance.

## 5 References

1. ISBSG data Benchmark Report Release 5 - <http://www.isbsg.org/html/index2.html>
2. Jamieson, D., Vinsen, K. and Callender, G. (2004) "Measuring Software Costs - A New Perspective on a Recurring Problem." Proceedings of the Sixth International Workshop on Economics-Driven Software Engineering. Edinburgh
3. Jamieson, D., Vinsen, K. and Callender, G. (2004) "Modelling Software Development Costs - New Approaches to an Old Problem". In Delener, N. & Chao, C. Navigating Crisis and Opportunities in Global Markets. GBATA, New York
4. Software Productivity Research (Capers Jones) - Software's Chronic Crisis W. Gibbs Scientific American September 1994
5. SouthernSCOPE Project Management Methodology developed by the Australian Government (State of Victoria). - <http://www.egov.vic.gov.au/index.php?env=-innews/detail.tpl:m1816-1-1-7:l0-0-1-:n832-0-0>