Luigi Buglione & Alain Abran, *Universitè du Québec à Montréal (UQÀM), Canada*

Multidimensionality in Software Performance Measurement: the QEST/LIME models

Abstract -- *A general weakness in problem analysis and problem solving is the oversimplification of the complex problems to be studied. For example, in software measurement and management, almost all performance measurement models are based on two-dimensional representations, such as histograms or the Kiviat graph. Through the application of geometrical concepts and techniques, we propose a new multi-dimensional model in this paper to measure the performance of software projects.*

Index Terms -- Software Performance Measurement, Geometry, Multidimensionality, QEST/LIME, Information Systems

I Introduction

In a competitive market period such as the current one, a company's success strongly depends on its ability to react on time to customer requests and minimize the cost of goods and services offered, as well as on efficiently meeting a myriad of other constraints. Monitoring these numerous constraints and their impact on the development process simultaneously is increasingly critical. Therefore, continuous measurement of performance becomes a key component in improving the planning, monitoring and delivery of goods and services, as well as for the design of improvement programs.

Performance is not a single, one-dimensional concept: it is not enough to meet a specific target in an unconstrained environment. Performance is a multidimensional concept which must integrate multiple viewpoints, most of which are present simultaneously in the software development process.

In this paper, we tackle this multidimensional representation issue by proposing an application of geometry for the measurement of the software projects performances. In section 2, a proposal of three key viewpoints for evaluating a project is discussed. In section 3, the previous attempts at using geometry to measure software project performance are summarized. In section 4, an improved solution is presented, based on a three-dimensional representation, the QEST/LIME model. Finally, a summary of observations is given in section 5.

Authors' references: Software Engineering Management Research Laboratory , Université du Québec à Montréal, C.P. 8888, Succ. Centre-Ville, Montréal, Québec, Canada. **Luigi Buglione** - E-mail: luigi.buglione@computer.org, Tel: (39) 338.95.46.917, Fax: (39) 06-233.208.366; **Alain Abran** - E-mail: abran.alain@uqam.ca, Tel: +1 (514) 987-3000 (8900), Fax: +1 (514) 987-8477.

II Recommended viewpoints for software projects

The ISO software engineering standards on software product quality, ISO/IEC 9126 and 14598 [ISO91] [ISO99], have identified three major types of stakeholders involved in software development projects: managers, users and developers. Each type of stakeholder represents a distinct viewpoint from which software performance can be evaluated; we have labeled these viewpoints as follows:

- **E**conomic: represented by the managers' viewpoint;
- **S**ocial: represented by the users' viewpoint;
- **T**echnical: represented by the developers' viewpoint.

Current performance models in the software engineering literature, however, only take into consideration either the first or third, or the first and third, of the viewpoints listed above; and, even when two viewpoints are taken into account, the current performance models handle them separately. In this paper, we present our proposal for a multidimensional performance model designed to tackle these three viewpoints simultaneously.

III How Geometry helps in representing complex realities

Analyzing a huge amount of project data from several viewpoints, but a single dimension at a time, makes it difficult to synthesize the information into an overall assessment for decision-making purposes.

Graphical representation can help in simplifying and increasing data readability for managers. In its simplest expression, the use of histograms, bars or pie charts is currently a well-known means for representing some aspects of more complex realities, through the use of simple statistical tools such as averages, medians or quartiles. In particular, it is now much easier to use a spreadsheet like MS Excel and select the "graph" tool in order to choose from among more than a dozen graphical representations. However, almost all these representations are two-dimensional. Moreover, they are often measured through two-dimensional concepts .

One example is provided by the Kiviat graph [KOLE73], first introduced in 1973 for evaluating the performance of computer systems by plotting the semi-axes radiating from a point, called a *pole*. Each aspect to be evaluated is represented by a semi-axis, and its performance level is plotted along the axes using predetermined scales. Then, all the points are connected, creating a 2D shape (polygon)[1].

While in practice most managers limit themselves to a visual analysis of the pattern on the Kiviat diagram, there of

[1] A well-balanced system should then produce a star-shaped graph. Another issue to note is that systems tend to produce characteristic shapes, as stressed by [FERR83], and a skilled system manager should recognize this fact, applying the proper adjustments.

course exist techniques to assign numerical values to such representations, through geometry and using the *distance* and *area* concepts[2]. For instance, overall performance can be calculated using the area of the polygon in the Kiviat diagram. Improvements across multiple project phases can be measured graphically by overlapping two subsequent Kiviat graphs and numerically by calculating the ratio between the two areas, as shown in the following figure:

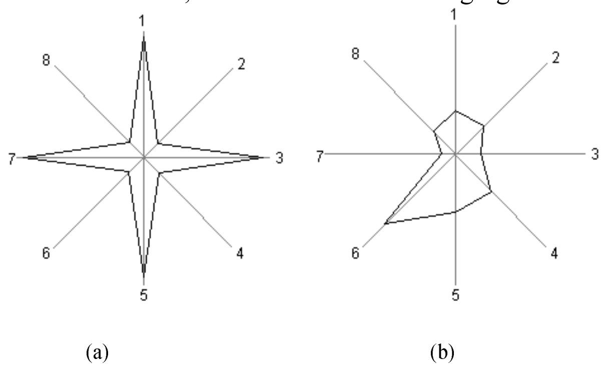

Fig. 1 – *Kiviat graphs: (a) well-balanced system; (b) untuned system*

While Kiviat diagrams contain rich visual information, they also have some weaknesses in terms of combining the information into single numerical values for overall assessment purposes. For instance, while they address the multi-element context, their representation is still a 2D one; for a successful use of geometrical analysis and the representation of performance from multiple viewpoints, there should be a direct relationship between the number of elements (or perspectives) and the dimensions in a geometrical space for a proper representation: 2 elements in a 2D space, 3 elements in a 3D space, and so on.

Principle: A correspondent relationship between the number of elements to represent and the dimension in which the representation format is expressed must exist.

Therefore, a representation such as that of Kiviat graphs which uses more than 2 elements in a 2D space represents a *compression* of a more complex reality. A quick and immediate example of this concept can be provided by the ancient Egyptian painting (Figure 2): the observer realizes that something is missing (the *depth* of the image), knowing that its source was a 3D figure.

Fig. 2 – *Egyptian painting: a 3D concept fitted into a 2D representation*

To tackle (at least) three viewpoints for a more comprehensive evaluation of the performance of a system (economic, social and technical), we initially identified three distinct approaches in the software engineering literature dealing with multidimensionality in software performance measurement:

- in Gonzales [GONZ95], a **vectorial approach** is proposed to measure software complexity in a 3D space; the three *dimensions* are given by length, time and level for each of the three complexity domains (syntactical, functional, computational) with a list of predefined and non-normalized complexity factors and measures;
- in Hatfield [HATF95], the measurement of product performance is defined as the single viewpoint related to product assessment (asset / customer-project / strategic management) as a *dimension*, but it represents the 3D concept by means of a **cube** and uses only a single non-normalized measure per dimension;
- in Donaldson & Siegel [DONA97], *n* different normalized measures are used to define the “product integrity value” (and not the single interest group) as a *dimension* using a vectorial approach, representing the concept in a 2D space through the use of **Kiviat graphs**.

While each approach is of interest, none fully addresses the simultaneity of the multidimensional representation of performance. We present next our solution for a multidimensional performance model (QEST) and its extension to the whole software life cycle development (LIME) using a classic geometrical approach.

IV The QEST/LIME models

In this section, we present our approach for a multidimensional representation of software project performance. The QEST[3] model allows us to use a certain number of measures not predetermined by the model itself. Furthermore, it allows for the use of normalized ranges of

[2] For instance, an educational project using plane geometry was "*Geomland*", headed by the Stevens Institute of Technology (USA) and the Bulgarian Academy of Sciences [FRIE97], with the following aim: to "*provide opportunities for students to experiment with geometric objects and concepts. [This software] uses animation of geometric construction in unique ways that make it possible to create simple visualizations of complex relationships and ideas.*"

[3] QEST**:** Quality Factor + Economic, Social and Technical viewpoints

values (unlike the Gonzales and Hatfield approaches) and expresses the performance measurement concept with a 3D construction (rather than a 2D one, as in Donaldson & Siegel), and with a pyramidal representation (3 sides - 3 viewpoints) rather than a cubic one (4 sides - 3 viewpoints, as in Hatfield); QEST proposes a geometrical representation of performance with the same number of sides as the number of viewpoints considered.

Another unusual feature of the QEST model is the following: the measurement of performance (**p**) is given by the *integration* of an instrument-based measurement process (expressed in the model by the component **RP** - *Rough Productivity*) with a measurement process based on the subjective perception of quality (expressed in the model by the component **QF** - *Quality Factor*).

In summary:

Performance = Productivity (PR) *and* Quality (Q)
Performance measurement = (Instrument-based measurement of Productivity) *and* (Perception-based measurement of Quality)

In the following sub-sections, the QEST model is presented first, followed by its extension to the whole software life cycle (SLC), namely the **LIME** (LIfecycle MEasurement) model.

The QEST model

The QEST model provides a multidimensional *structured shell* which can then be populated with the criteria specified for any project according to management objectives: it is therefore referred to as an *open model*. This topology of performance models also makes it possible to handle the multiple and distinct viewpoints already discussed, all of which can exist concurrently in any software project. Several publications present specific characteristics and features of the QEST model in greater depth, such as:

- Theoretical considerations [BUGL99a] [BUGL99b]
- Geometrical and statistical foundations [BUGL99c]
- Implementation of the model [BUGL01c]
- Quality Factor [BUGL99b] and Quality Factor through QFD [BUGL01a].

In this section, we present the high-level design of this *open model* for the measurement of software project performance. The basic purpose of the structured shell of the open model is, as stated above, to express performance as the combination of the specific measures (or sets of measures) selected for each of the three dimensions, these values being derived from both an instrument-based measurement of productivity and a perception-based measurement of quality.
A three-dimensional geometrical representation of a *regular tetrahedron* was selected as the basis for the model, and is illustrated in Figure 3. Furthermore:

- the three dimensions (E, S, T) in the space correspond to the corners of the pyramid's base, and the convergence of the edges to the P vertex correspond to the maximum performance level;
- when the three sides are of equal length, the solid shape that represents this three-dimensional concept is therefore a pyramid with its triangular base and sides of equal length (*tetrahedron*).

This pyramid-type representation imposes the following constraint: the sides must be equal. In practice, this constraint is met by giving equal weight to each of the three different dimensions chosen – and with sides of length exactly equal to 1 (*regular tetrahedron*); this then allows representation of the dimensions through a normalized set of values between 0 and 1 for each axis and on a ratio scale, for ease of understanding.

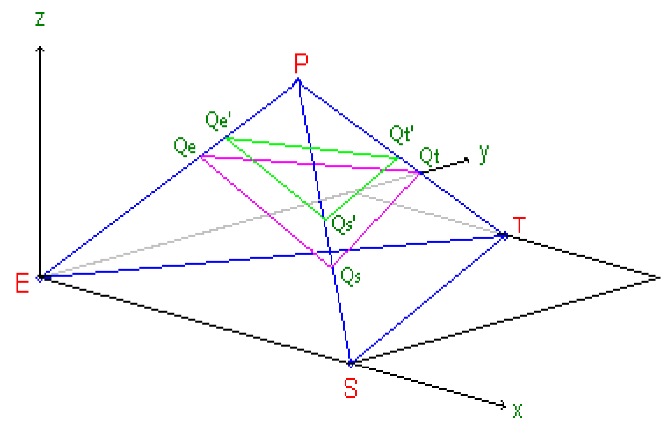

Figure 3 - *QEST model and its hyperplane sections*

With this 3D representation, it is possible to determine and represent performance using the three classic geometrical concepts of distance, area and volume; furthermore, the ratio between the volume of the lower part of the truncated tetrahedron and the total volume of the tetrahedron itself represents the normalized performance level of a project being assessed against its performance target, as demonstrated in [BUGL99c].

Thus, the volume within this 3D shape immediately shows in a visual way to what extent the tetrahedron is filled, and, when performance is re-assessed throughout a project life cycle, the sloped section represents how performance is progressing among the three dimensions. This is the added value that geometry can provide. Using an analogy, a management dashboard's purpose is to have available a series of gauges and controls to summarize information and to provide a feel for the status of a certain phenomenon. However, the dashboard can only properly represent 2D concepts; the insertion of a third variable requires more 2D controls or a new way to represent it, such as the one proposed in the QEST model.

Key features of the QEST model:

- **Integrated quantitative and qualitative evaluation from three concurrent organizational viewpoints:** management (*economic* viewpoint), users (*social* viewpoint) and technical personnel (*technical* viewpoint). Performance is not a single, one-dimensional concept. Hence, it is not enough to meet a single, specific target in an unconstrained environment. Performance is a multidimensional concept in which multiple viewpoints must be integrated, most of which are present simultaneously in the software development process. The other fundamental requirement is the integration of qualitative and quantitative evaluations, intended as a 2-sided face of the same problem. Only if these dimensions are managed concurrently in an integrated model for performance measurement can they be adequately assessed;

- **Use of de facto and de jure standards,** such as ISO/IEC Standard 9126 on Software Quality Characteristics and Sub-characteristics and, more recently, the ISO software functional-size-related measures. The use of such measurement standards is strongly recommended because they give the international software engineering community the opportunity to share and use the same definitions of objects and concepts in their work, reducing the potential for misinterpretation and miscommunication. For this reason, both de jure (ISO/IEC 9126) and de facto[4] standards are recommended when implementing a performance model;
- **A 3D geometrical construction** – The geometrical approach permits representation of the measurement of performance in a simple and visual way for an immediate grasp of the current status of the performance of a project at any time. The original selection of the regular tetrahedron was also suggested by the idea that the vertex of the 3D shape represents, from a conceptual viewpoint, the *convergence* of different viewpoint evaluations into a final, single one. Another important factor to take into account is the use of normalized values in order to give management greater value readability for taking decisions.

The LIME model

The LIME [BUGL99] [BUGL00] model extends the QEST model concepts to a dynamic context, such that the model can be applied to each step of any topology of development model selected. For illustrative purposes, the LIME model considers a generic 6-phase waterfall development model in Figure 4. Using a process notation such as ETVX (Entry-Task-Validation-eXit) [RADI88], the results of a QEST assessment at a particular time for the (n-1)th phase represents the input for the nth one; processing produces the nth output, which will be the input for the (n+1)th phase, and so on. The framework for the LIME model is the following:

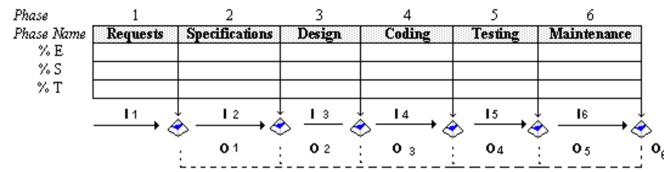

Fig. 44 – *The LIME model*

The iterative definition, collection and analysis of multidimensional measures at each life cycle phase therefore offers the feedback required to make adjustments to the project processes in a timely fashion, both for the next phase and for designing future improvements to the process of the preceding phase. This is illustrated in Figure 5 with the four process improvement steps of the PMAI (Plan – Measure – Analyze – Improve) cycle for each of the six phases of the waterfall model considered, included in every phase on a complete QEST assessment[5].

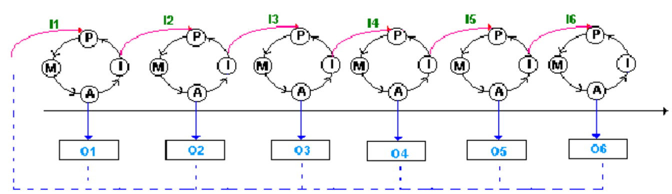

Fig. 5 – *LIME model and PMAI cycle*

Thus, performance is reassessed at each project phase, and, over the course of a project, a series of tetrahedrons will be drawn and the sequence of volumes will indicate the progression of the project performance, step by step, according to the three viewpoints.

Evolution of the QEST/LIME model

Of course, other profiles could be included in a QEST/LIME model, including quality models such as EFQM or the Malcolm Baldridge models, or others like the Balanced Scorecard or the Skandia Navigator where the number of stakeholders can be significantly increased.

How can we then properly apply the QEST/LIME models to more than three viewpoints?

The only way to represent and calculate joint assessments from more than three viewpoints is through algebra and not by using a geometrical approach. It was demonstrated in [BUGL01b] that the mechanism to be used for n-dimensions is a *simplex*, definable as the convex hull of any $d+1$ affinely independent points in some $\Re^n$ (n>d), where d represents the number of dimensions taken into account. There are, of course, some strengths and weaknesses in this process. For instance, by using matrix calculation, it is possible to extend the underlying concepts of the QEST/LIME models to n dimensions, and we have demonstrated the back-forward validity of this mechanism for any number of dimensions. The models are thus able to take into account a greater number of stakeholders.

However, from the fourth dimension on, it is no longer

---

[4] While work on ISO 14143 [ISO98] is under way and the Functional Size Measurement Method (FSM) is being standardized by the International Organization on Standardisation, the COSMIC initiative (http://www.cosmicom.com) is developing and promoting the next generation of functional size measurement methods.

[5] For further details about the PMAI cycle, its steps and activities, refer to [BUGL01b]

possible to take advantage of geometry or the visualization of related concepts in a *direct* way.

Increasing the number of stakeholders and fixing the dimensionality of the representation schema is equivalent to what the Egyptians did in the painting shown in Fig.2: the observer can only look at a flat image, where there is too much information to properly represent the 3D image, and the loss of this information is evident. Thus, this n-dimensional extension, called *QEST nD*, offers the possibility of extending the same measurement idea and of proposing it again, according to the "dimensional" principle expressed in section 2.

V Conclusions

A good visual representation of information provides analysts and managers with more direct meta-information on project data. Dashboard and Kiviat graphs are illustrations of such types of visual information. Because of the greater number of viewpoints that should be included in performance assessment, a three-dimensional representation is required when at least the economic, social, and technical viewpoints are to be taken into account together in the assessment of any development project.

The QEST model is a new way to look at the application of geometry for evaluating assessment results: the shape chosen was a regular tetrahedron, since it is the only geometrical shape with a single vertex where all the other opinions converge from the base. QEST was designed to be applied to ex-post or single project phase evaluations: from this point, the LIME model has been created in order to extend the same concept to the whole project.

While the QEST model incorporates two kinds of evaluations, quantitative and qualitative, these components can also be used as separate parts for partial assessments, as indicated by the following table.

QEST	**QEST nD**	**QF**	**QF^{2}D**	**LIME**	**Scope**					
					[1] Quant	**[2] Qual**	**[3] Single phase**	**[4] All Phases**	**[5] 3 dim**	**[6] n dim**
*		*			*	*	*		*	
*			*		*	*	*		*	
	*	*			*	*	*			*
	*		*		*	*	*			*
*					*		*		*	
	*				*		*			*
		*				*	*		*	
			*			*		*		*
*		*		*	*	*		*	*	
*			*	*	*	*		*		*

The "Scope" column has been subdivided into 6 sub-columns:

- **[1] Quant**: Quantitative assessment
- **[2] Qual**: Qualitative assessment
- **[3] Single phase**: the selected method has been applied just to a single project phase
- **[4]All phases**: the selected method has been applied to all the project phases
- **[5] 3 dim**: the selected method is applicable to 3 viewpoints
- **[6] n dim**: the selected method/technique is applicable to n viewpoints

Thus, applying the basic QEST model (fifth row) means working on a quantitative assessment [1] for a single project phase [3] dealing with three viewpoints [5]; applying the complete QEST nD means working on a quantitative + a qualitative assessment, where the qualitative component [1+2] can be QF (in the case of the third row) or QF^{2}D (fourth row) dealing with n possible viewpoints [6]; and so on.

The final suggestion is that an attempt should be made to generalize, whenever possible and useful, recourse to geometry in the analysis of complex problems and in their representations in new knowledge areas, such as Software Engineering Management (the Knowledge Area in which Software Measurement is included [SWEBOK01]).

References

[BUGL99a] BUGLIONE L. & ABRAN A., *Multidimensional Software Performance Measurement Models: A Tetrahedron-based Design*, in "Software Measurement: Current Trends in Research and Practice", R. Dumke/A. Abran (eds.), Deutscher Universitats Verlag GmbH, pp. 93-107, 1999, ISBN 3-8244-6876-X

[BUGL99b] BUGLIONE L. & ABRAN A., *A Quality Factor for Software*, Proceedings of QUALITA99, 3rd International Conference on Quality and Reliability, Paris, France, 25-26 March 1999, pp- 335-344.

[BUGL99c] BUGLIONE L. & ABRAN A., *Geometrical and Statistical Foundations of a Three-dimensional Model of Performance*, "International Journal of Advances in Engineering Software", Elsevier Publisher, Vol. 30 No. 12, December 1999, pp. 913-919.

[BUGL99d] BUGLIONE L., *Misurare il software. Quantità, qualità, standard e miglioramento di processo nell'Information Technology*, Franco Angeli, ISBN 88-464-1729-1, 1999.

[BUGL00] BUGLIONE L. & ABRAN A, *LIME: A Three-Dimensional Software Performance Measurement Model for Project Management*, 2nd World Conference on Software Quality, September 25-29, 2000, Yokoyama, Tokyo, Japan

[BUGL01a] BUGLIONE L. & ABRAN A., *QF^{2}D: a different way to measure Software Quality*, included in: New Approaches in Software Measurement, 10th International Workshop on Software Measurement, IWSM 2000, Berlin, Germany, October 4-6, 2000. Proceedings, Dumke R. & Abran A. (Eds.), February 2001, ISBN 3-540-41727-3

[BUGL01b] BUGLIONE L. & ABRAN A., *QEST nD: n-dimensional extension and generalisation of a Software Performance Measurement Model,* submitted for publication to "International Journal of Advances in Engineering Software", Elsevier Publisher, 2001

[BUGL01c] BUGLIONE L. & ABRAN A., *Implementation of a Multidimensional Software Performance Model*, Université du Québec à Montréal, Technical Report, (to be published), 2001

[DONA97] DONALDSON, S.E. & SIEGEL, S.G., *Cultivating Successful Software Development: A Practitioner's View*, Prentice Hall, 1997.

[FERR83] FERRARI D., SERAZZI G. & ZEIGNER A., *Measurement and Tuning of Computer Systems,* Prentice-Hall, Englewood Cliffs, NJ, 1983

[FRIE97] FRIEDMAN E., *Bulgarian Transformation Geometry Educational Project*, 1997, URL:

http://www.bestpraceduc.org/DiscoveryGrants1997/BulgarianGeometry.html

[GONZ95] GONZALEZ, R.R., *A Unified Metric of Software Complexity: Measuring Productivity, Quality and Value*, The Journal of Systems and Software, Elsevier Science North-Holland, Vol. 29, No. 1, April 1995, 17-37.

[HATF95] HATFIELD, M.A., *Managing to the Corner Tube: Three-Dimensional Management in a Three-Dimensional World*, IEEE Engineering Management Review, IEEE Computer Society, Vol. 23, No. 3, Winter 1995, 63-68.

[ISO91] ISO/IEC, *International Standard 9126: Information Technology – Software product evaluation – Quality characteristics and guidelines for their use,* 1991

[ISO98] ISO/IEC JTC1/SC7/WG12, *IS 14143-1: Information Technology – Software Measurement - Functional Size Measurement – Part 1: Functional Size Measurement Concepts, 1998*

[ISO99] ISO/IEC JTC1/SC7/WG6, *IS 14598-1: Information Technology – Software product evaluation – Part 1: General Overview,* 1999

[KOLE73] KOLENCE K.W. & KIVIAT P J, *Software Unit Profiles and Kiviat Figure,* ACM SIGMETRICS Performance Evaluation Review 2(3) September 1973, pp.2-12

[RADI88] RADICE, R. A. & PHILLIPS, R.W., *Software Engineering: An Industrial Approach*, Prentice-Hall, 1988

[SWEBOK01] IEEE/ACM, *Guide to the Software Engineering Body of Knowledge,* Trial Version 0.95, May 2001, URL: http://www.swebok.org

[WILS93] WILSON J.W., FERNANDEZ M.L. & HADAWAY N.., *Mathematical Problem Solving*, Chapter 4 included in *Research Ideas for the Classroom: High School Mathematics*, Wilson P.S. (Eds), NY, MacMillian, 1993