

## Chapter 42

# Use of Qualitative Research to Generate a Function for Finding the Unit Cost of Software Test Cases

**Mark L. Gillenson**

*University of Memphis, Memphis, USA*

**Thomas F. Stafford**

*Louisiana Tech University, Ruston, USA*

**Xihui “Paul” Zhang**

*University of North Alabama, Florence, USA*

**Yao Shi**

*University of Memphis, Memphis, USA*

### **ABSTRACT**

*In this article, we demonstrate a novel use of case research to generate an empirical function through qualitative generalization. This innovative technique applies interpretive case analysis to the problem of defining and generalizing an empirical cost function for test cases through qualitative interaction with an industry cohort of subject matter experts involved in software testing at leading technology companies. While the technique is fully generalizable, this article demonstrates this technique with an example taken from the important field of software testing. The huge amount of software development conducted in today’s world makes taking its cost into account imperative. While software testing is a critical aspect of the software development process, little attention has been paid to the cost of testing code, and specifically to the cost of test cases, in comparison to the cost of developing code. Our research fills the gap by providing a function for estimating the cost of test cases.*

DOI: 10.4018/978-1-6684-3702-5.ch042

## INTRODUCTION

This is a demonstration of case research used to generate an empirical function, and it is an unusual contribution to the literature in that respect. Mathematical expressions of functionality are usually developed statistically, supported with survey or simulation data (Akhavein et al., 1997; Kurfman et al., 2003). It is rare for a case study to be quantitative, yet mathematical functions can (conceivably) be derived in this way since deductions involving mathematical propositions are a subset of formal logic and nothing in formal logic is tied to a specific methodology (Lee, 1989). The evidence that is considered in case studies can be qualitative, quantitative, or both (Eisenhardt, 1989), but the primary purpose of a case study is to generate theory (Yin, 2013) and a theoretical representation of cause and effect (as is typically found in a statement of mathematical logic, such as a response function) is not out of the question in case research, just rare (Lee, 1989). It is, practically speaking, more likely that response equations would lead to case studies (Hengl et al., 2007), since many statistical techniques benefit greatly from a qualitative interpolation process. In contrast to the conventional wisdom on case study uses and outcomes, however, we utilize an in-depth case study of industry experts to directly induce an empirical function through qualitative generalization (e.g., Yin, 2013).

Our methodology demonstrates the derivation of the variables and interrelated functionality of a candidate empirical expression. This methodology can be used in any situation in which there is a need to develop an empirical expression from case-based, qualitative research. While we will describe our innovative methodology in general terms, we believe that a detailed exposition of its actual use is essential to both understanding the process and to demonstrating its worth.

The actual example we have chosen is that of developing a function to estimate the cost of developing and executing a test case in the software testing process. The outcome of this process results in the description of the cost of generating and utilizing a functional test case for software testing purposes; this outcome of our case research is much in the way that Yin (2013) and Eisenhardt (1989) each suggests extracting theoretical meaning from case research. In that sense, the contribution of this study lies in generalizing from qualitative inquiry to a theoretical construction, as Yin and Eisenhardt would put it. For clarity, this will necessitate our providing some explanation of the elements of generating and executing software test cases, and of the software testing process, itself.

Case studies, concentrated as they are on a focused and bounded phenomenon embedded in a context, are particularly useful for identifying context-specific meaning (Eisenhardt, 1989; Miles et al., 2013). We consider that case study methods are typically intended to lead to theoretical developments as their own unique modality of generality (Yin, 2013), rather than the broad extension of generality to other contextual areas as is so typically the case with quantitative research. In that manner, we consider this in-depth investigation of software testing cost factors to be a revelatory approach (Sarker et al., 2012; Yin, 2013) meant for specific and detailed understanding of a unique context.

In addition, one of the most important potential theoretical contributions that could be made in the software development literature would be a functional model useful for determining the costs components of software testing, and this is an additional contribution of this study: the derivation of an empirical cost function as a generalization from an interpretive case study of software testing engineers brought together in a cohort as part of a certification program on software testing.

23 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/use-of-qualitative-research-to-generate-a-function-for-finding-the-unit-cost-of-software-test-cases/294498](http://www.igi-global.com/chapter/use-of-qualitative-research-to-generate-a-function-for-finding-the-unit-cost-of-software-test-cases/294498)

## Related Content

---

### Modeling Platform-Independent and Platform-Specific Service Architectures with UML and the ArchiMeDeS Framework

Marcos López-Sanz and Esperanza Marcos (2014). *Handbook of Research on Emerging Advancements and Technologies in Software Engineering* (pp. 254-277).

[www.irma-international.org/chapter/modeling-platform-independent-and-platform-specific-service-architectures-with-uml-and-the-archimedes-framework/108619](http://www.irma-international.org/chapter/modeling-platform-independent-and-platform-specific-service-architectures-with-uml-and-the-archimedes-framework/108619)

### SQL Scorecard for Improved Stability and Performance of Data Warehouses

Nayem Rahman (2016). *International Journal of Software Innovation* (pp. 22-37).

[www.irma-international.org/article/sql-scorecard-for-improved-stability-and-performance-of-data-warehouses/157277](http://www.irma-international.org/article/sql-scorecard-for-improved-stability-and-performance-of-data-warehouses/157277)

### Numerical Simulation of Distributed Dynamic Systems using Hybrid Tools of Intelligent Computing

Fethi H. Bellamine and Aymen Gdouda (2013). *Embedded Computing Systems: Applications, Optimization, and Advanced Design* (pp. 360-383).

[www.irma-international.org/chapter/numerical-simulation-distributed-dynamic-systems/76965](http://www.irma-international.org/chapter/numerical-simulation-distributed-dynamic-systems/76965)

### Construction of Fuzzy Database and Analysis Interface Using Fuzzy Graphs for Management System Operation Analysis

Yasunori Shiono, Toshihiro Yoshizumi, Takaaki Goto, Tadaaki Kirishima and Kensei Tsuchida (2022). *International Journal of Software Innovation* (pp. 1-16).

[www.irma-international.org/article/construction-of-fuzzy-database-and-analysis-interface-using-fuzzy-graphs-for-management-system-operation-analysis/307014](http://www.irma-international.org/article/construction-of-fuzzy-database-and-analysis-interface-using-fuzzy-graphs-for-management-system-operation-analysis/307014)

### Innovation Process for Problem Conceptualization

Sharon Andrews (2021). *International Journal of Software Innovation* (pp. 91-103).

[www.irma-international.org/article/innovation-process-for-problem-conceptualization/298971](http://www.irma-international.org/article/innovation-process-for-problem-conceptualization/298971)