

# Chapter 14

## Code Reuse

**Donna Bridgham**  
DARS, USA

### ABSTRACT

*The reuse of code can be used to add or update functionalities with little or no modifications to new or existing software applications. Developers have reused sections of code when the code is available but have been hindered by finding the code that is needed for an application. By creating a code repository, code would be available to developers in a systemic method. The code would be available for functional and nonfunctional uses in applications. Since the code has already be written, during the discovery phase of projects the developers involved should be able to search the repository for the code that is needed for strategies and problems that have already been successfully been implemented. Quality, cost, and time should be the focus of code reuse. To maximize code reuse, a code repository that is properly categorized and indexed would add to the software development lifecycle by making code available to developers that they can use with confidence. The code repository will improve the application process.*

### INTRODUCTION

The reuse of code can be used to add or update functionalities with little or no modifications to new or existing software applications such as including media in the development. Developers have reused sections of code when the code is available but have been hindered by finding the code that is needed for an application or specific action. By creating a code repository, code would be available to developers in a systemic method.

With the use of web applications this makes the use of a repository useful for all types of media. Since many web applications reuse the same style sheets, classes and modules they can be held in a repository. Since many languages are used, code libraries are no longer enough. A code repository would solve the problem by have the code available.

The code would be available for functional and nonfunctional uses in applications. Since the code has already been written, during the discovery phase of software projects the developers involved should be able to search the repository for the code that is needed for strategies and problems that have already been successfully been implemented. Quality, cost, and time should be the focus of code reuse.

DOI: 10.4018/978-1-5225-0648-5.ch014

To maximize code reuse, a code repository needs to be properly categorized and indexed so it would add another dimension to the software development lifecycle. If the code is made available to developers through the organizations own secure web site and the code is monitored the code repository can be used by the developers with confidence. The developers will be able to share code from previous applications and media designed by previous developers that has already been tested and used in existing applications so there will be a higher confidence level. This would be a strong start to creating a code repository.

Since the code had already been implemented in previous releases, the code will be available for the developers to find since it will be catalogued and indexed. Also by sharing code, developers will be encouraged to write comments in their code which will help other developers understand the purpose of the code. The code repository will improve the application process for the research purpose.

After implementation, all developers will be able to access the Code Repository through a web interface. It will have authentication so only approved personnel can look at the repository. There will be a screen where the developer can look for the category they need, the problem or action they are looking for and then a snippet will appear for them to use. All associated bugs, fixes and additional data will be included. This will enable the developer to understand previous testing, previous uses and the actual intent of the code.

## BACKGROUND

Code reuse is not a new idea but moving it into a code repository is a new concept. Code reuse has been a practice for developers since it can be more effective, has fewer bugs and has been tested in previous versions of applications. If code is used on one application and then another application has the need for the same type of action or event then the code can be reused if other developers know the code is available (National Instruments, 2010). Since the code has already been tested, consolidated and meets the organizational standards the code should be reused. The more the code is reused and tested the less probability for bugs or errors (National Instruments, 2010). If code is reused it may also benefit the user since the action or event will be familiar. Researchers will enjoy new features but will also identify with familiarity that will help with improving research. Since developers already reuse code, the next logical step is to build a code repository.

When a developer is writing code for an application and they are able to find code that has already been used and tested, it will reduce the risk of error and speed up the development and research process making an efficient process. The risk therefore is not in the code that is reused since the probability of bugs or failure will most likely be in the new lines of codes so it will be easier to find the errors (National Instruments, 2010). It is important to find a place to store and use code in an area that all developers can access. A repository would be an efficient place to store the code. The repository can be a web application frontend with a file server backend or database backend. This can be secured so only authorized personnel can access the repository.

There are problems with reusing codes and this also must be addressed. When integrating code it should always be analyzed before integrating it into the project (Schmidt, 2012). The code should be evaluated on several levels, such as which code is the best solution, have all previous bugs or errors been addressed, and will any modifications cause a change in the integrity of the code. The reuse of the code should save money and not cost more money so an evaluation should be done to ensure that the reuse is

17 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/code-reuse/161973](http://www.igi-global.com/chapter/code-reuse/161973)

## Related Content

---

### Sentiment Analysis of Brand Personality Positioning Through Text Mining

Ruei-Shan Lu, Hsiu-Yuan Tsao, Hao-Chaing Koong Lin, Yu-Chun Maand Cheng-Tung Chuang (2022). *Research Anthology on Implementing Sentiment Analysis Across Multiple Disciplines* (pp. 852-863).

[www.irma-international.org/chapter/sentiment-analysis-of-brand-personality-positioning-through-text-mining/308523](http://www.irma-international.org/chapter/sentiment-analysis-of-brand-personality-positioning-through-text-mining/308523)

### Reliability Estimates for Regression Predictions: Performance Analysis

Zoran Bosnicand Igor Kononenko (2011). *Integrations of Data Warehousing, Data Mining and Database Technologies: Innovative Approaches* (pp. 320-338).

[www.irma-international.org/chapter/reliability-estimates-regression-predictions/53082](http://www.irma-international.org/chapter/reliability-estimates-regression-predictions/53082)

### Logical and Physical Design of Spatial Non-Strict Hierarchies in Relational Spatial Data Warehouse

Ferrahi Ibtisam Ibtisam, Sandro Bimonteand Kamel Boukhalfa (2019). *International Journal of Data Warehousing and Mining* (pp. 1-18).

[www.irma-international.org/article/logical-and-physical-design-of-spatial-non-strict-hierarchies-in-relational-spatial-data-warehouse/223134](http://www.irma-international.org/article/logical-and-physical-design-of-spatial-non-strict-hierarchies-in-relational-spatial-data-warehouse/223134)

### Data Discovery Over Time Series From Star Schemas Based on Association, Correlation, and Causality

Wallace Anacleto Pinheiro, Geraldo Xexéo, Jano Moreira de Souzaand Ana Bárbara Sapienza Pinheiro (2020). *International Journal of Data Warehousing and Mining* (pp. 95-111).

[www.irma-international.org/article/data-discovery-over-time-series-from-star-schemas-based-on-association-correlation-and-causality/265259](http://www.irma-international.org/article/data-discovery-over-time-series-from-star-schemas-based-on-association-correlation-and-causality/265259)

### Approximate Range Querying over Sliding Windows

Francesco Buccafurri (2009). *Handbook of Research on Text and Web Mining Technologies* (pp. 273-287).

[www.irma-international.org/chapter/approximate-range-querying-over-sliding/21729](http://www.irma-international.org/chapter/approximate-range-querying-over-sliding/21729)