

発表概要

# ライブプログラミング環境における ユニットテスト機能の設計と実現方法

今井 朝貴<sup>1,a)</sup> 増原 英彦<sup>2,b)</sup> 青谷 知幸<sup>2,c)</sup>

2016年10月27日発表

ライブプログラミング環境はプログラム中の式をコード編集中に実行し、即座にその結果を表示する。この即時のフィードバックにより、プログラマは一時な式をプログラム中に書き、引数や定義を変更することで試行錯誤的なプログラミングを容易に行える。我々は試行錯誤的なプログラミング過程とテスト駆動開発の類似性に注目し、ライブプログラミングのためのユニットテスト機能を提案する。この機能は、(1) 一時的な式とユニットテストケースを簡単な相互変換し、(2) 実行途中に得られた値を式に変換し、ユニットテストの期待値として利用可能にする。我々はこのテスト機能を Shiranui ライブプログラミング環境中に作成した。本発表では、このテスト機能の設計とインタプリタによる実現方式を紹介する。

## Design and Implementation of Unit Testing Support in a Live Programming Environment

TOMOKI IMAI<sup>1,a)</sup> HIDEHIKO MASUHARA<sup>2,b)</sup> TOMOYUKI AOTANI<sup>2,c)</sup>

Presented: October 27, 2016

Live programming environments execute expressions in a program during code editing, and instantly show their results. With this instant feedback mechanism, the programmer can easily perform exploratory programming by writing transient expressions, and changing parameters or program definitions. We focus on the similarity between the exploratory programming process and test-driven development, and propose a novel unit testing feature for live programming. The feature (1) seamlessly converts a transient expression and a unit test case, and (2) extracts an intermediate execution result into an expression so that it can be used as an expected value in a unit test case. We realized this feature in our live programming environment called Shiranui. In this presentation, we present the design of those features, and its interpreter-based implementation.

<sup>1</sup> 東京工業大学大学院情報理工学研究科数理・計算科学専攻  
Department of Mathematical and Computing Sciences,  
Tokyo Institute of Technology, Meguro, Tokyo 152-8552,  
Japan

<sup>2</sup> 東京工業大学情報理工学院数理・計算科学系  
Department of Mathematical and Computing Sciences,  
Tokyo Institute of Technology, Meguro, Tokyo 152-8552,  
Japan

a) imai.t.af@m.titech.ac.jp

b) masuhara@acm.org

c) aotani@is.titech.ac.jp