ARTENOLIS: Automated Reproducibility and Testing Environment for Licensed Software

Laurent Heirendt\textsuperscript{1}& Sylvain Arreckx\textsuperscript{1}, Christophe Trefois\textsuperscript{1}, Yohan Jarosz\textsuperscript{1}, Maharshi Vyas\textsuperscript{1}, Venkata P. Satagopam\textsuperscript{1}, Reinhard Schneider\textsuperscript{1}, Ines Thiele\textsuperscript{1}, and Ronan M.T. Fleming\textsuperscript{1}*(ronan.mt.fleming@gmail.com)

\textsuperscript{1}Luxembourg Centre for Systems Biomedicine, University of Luxembourg, Esch-sur-Alzette, Luxembourg

Project Goals: The aim of this project is to develop a continuous integration framework to automatically test changes to code, evaluate their impact on an open-source code base with licensed dependencies, and ensure the stability, integrity, and cross-platform compatibility of the COBRA Toolbox.

Automatically testing changes to code is an essential feature of continuous integration. For open-source code, without licensed dependencies, a variety of continuous integration services exist. The COnstraint-Based Reconstruction and Analysis (COBRA) Toolbox \cite{1} is a suite of open-source code for computational modelling with dependencies on licensed software. It is distributed as open-source code, but as it is dependent on the licensed software MATLAB (The Mathworks, Inc.). A novel automated framework of continuous integration in a semi-licensed environment is required for the development of the COBRA Toolbox and related tools of the COBRA community. Here, we present ARTENOLIS \cite{2}, a freely accessible under [http://artenolis.lcsb.uni.lu](http://artenolis.lcsb.uni.lu) a general-purpose infrastructure software application that implements continuous integration for open-source software with licensed dependencies. It uses a master-slave framework, tests code on multiple operating systems, and multiple versions of licensed software dependencies. ARTENOLIS ensures the stability, integrity, and cross-platform compatibility of code in the COBRA Toolbox and related tools.

References


This work was supported by the U.S. Department of Energy, Offices of Advanced Scientific Computing Research and the Biological and Environmental Research as part of the Scientific Discovery Through Advanced Computing program, grant #DE-SC0010429, the National Centre of Excellence in Research, and the Luxembourg National Research Fund (FNR) ATTRACT program (FNR/A12/01) and OPEN program (FNR/O16/114/02054) grants.