Deep Software Variability: Towards Handling Cross-Layer Configuration

Luc Lesoil*[1]

[1]Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA) – Universite de Rennes 1 : UMR6074, Université de Rennes, Institut National des Sciences Appliquées - Rennes, Institut National des Sciences Appliquées, Université de Bretagne Sud, École normale supérieure - Rennes, Institut National de Recherche en Informatique et en Automatique, CentraleSupélec, Centre National de la Recherche Scientifique, IMT Atlantique Bretagne-Pays de la Loire, Institut Mines-Télécom [Paris] – Avenue du général LeclercCampus de Beaulieu 35042 RENNES CEDEX, France

Résumé

Configuring software is a powerful means to reach functional and performance goals of a system. However, many layers (hardware, operating system, input data, etc.), themselves subject to variability, can alter performances of software configurations. For instance, configurations' options of the x264 video encoder may have very different effects on x264's encoding time when used with different input videos, depending on the hardware on which it is executed. In this vision paper, we coin the term deep software variability to refer to the interaction of all external layers modifying the behavior or non-functional properties of a software. Deep software variability challenges practitioners and researchers: the combinatorial explosion of possible executing environments complicates the understanding, the configuration, the maintenance, the debug, and the test of configurable systems. There are also opportunities: harnessing all variability layers (and not only the software layer) can lead to more efficient systems and configuration knowledge that truly generalizes to any usage and context.

*Intervenant