
On the Benefits and Limits of Incremental Build of Software Configurations: An Exploratory Study

Georges Aaron Randrianaina*¹

¹Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA) – Université de Rennes, Institut National des Sciences Appliquées - Rennes, 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 – Avenue du général LeclercCampus de Beaulieu 35042 RENNES CEDEX, France

Résumé

Software projects use build systems to automate the compilation, testing, and continuous deployment of their software products. As software becomes increasingly configurable, the build of multiple configurations is a pressing need, but expensive and challenging to implement. The current state of practice is to build independently (*a.k.a.*, clean build) a software for a subset of configurations. While incremental build has been studied for software evolution and relatively small changes of the source code, it has surprisingly not been considered for software configurations. In this exploratory study, we examine the benefits and limits of building software configurations incrementally, rather than always building them cleanly. By using five real-life configurable systems as subjects, we explore whether incremental build works, outperforms a sequence of clean builds, is correct *w.r.t.* clean build, and can be used to find an optimal ordering for building configurations. Our results show that incremental build is feasible in 100% of the times in four subjects and in 78% of the times in one subject. In average, 88.5% of the configurations could be built faster with incremental build while also finding several alternatives faster incremental builds. However, only 60% of faster incremental builds are correct. Still, when considering those correct incremental builds with clean builds, we could always find an optimal order that is faster than just a collection of clean builds with a gain up to 11.76%.

*Intervenant