Code Siblings: Phenotype Evolution

Daniel M. German†, Massimiliano Di Penta‡, Giuliano Antoniol*, and Yann-Gaël Guéhéneuc*
dmg@uvic.ca,dipenta@unisannio.it,antoniol@ieee.org,yann-gael.gueheneuc@polymtl.ca

† Department of Computer Science, University of Victoria, BC, Canada
‡ Department of Engineering, University of Sannio, Benevento, Italy
* SOCCER Lab.–PTIDEJ Team, DGIGL, École Polytechnique de Montréal, QC, Canada

Our position was inspired by Philip K. Dick's book "Do Androids Dream of of Electric Sheeps", which became Ridley Scott's acclaimed movie "Blade Runner". Rachael Tyrell and Roy Batty are androids, *i.e.*, replicants of the same series. Rachel is a caring woman who does not know that she is a replicant while Roy is a slave who kills in an attempt to revert a failsafe system implanted in replicants to limit their life-span to four years. Both are clones of a same series, yet become very different individuals.

Our position focuses on the code migration between different software systems and the subsequent evolution of code clones. A piece of code—often an entire file or function—can be copied from one system to another for many different reasons, including adding features already implemented in the other system, the need to fix a bug relying on a known and robust implementation, or the migration of a developer from one project to another.

Our goal is to detect clones across systems, study the evolution of the copies under different environmental conditions, and determine the characteristics of clones that are mostly hereditary or environmental. In nature, an analogy can be found in species that, under the pressure of the environment, evolve towards new phenotypes. Much as Rachel and Roy had both to adapt to live in a civilized city or combat in off-world battles, code fragments (*e.g.*, operating system drivers) must be able to adapt to specific operating system constraints and requirements.

When clones are created (by copying and adapting code from one system to another) their environment changes. This is true between subsystems, however it is even worse across different systems, that can be different in terms of: (i) users, and user needs and market pressure; (ii) development teams; (iii) architectural and platform constraints, etc. Godfrey and German use the term *software phenotype* to refer to a program deployed within a specific environment [1]. We extend this notion to clones: the clone genotype is the code that is originally cloned; the clone phenotypes are the result of the evolution of their genotype when exposed to different environments.

To illustrate our point, we will use the Adaptec aic7XXX SCSI driver for Linux. We have discovered that this driver and those from FreeBSD, OpenBSD and NetBSD originated from the same code (circa 1995). Some parts of the driver were originally developed for Linux and others for FreeBSD. The development was then centralized and an effort was made to maintain a single source (using #ifdefs to manage the differences). In time they split again into Linux and FreeBSD versions, while OpenBSD and NetBSD continued to evolve their drivers by closely following the FreeBSD version. It appears that the Linux driver is the one with the largest number of users, and also the one with the most active development. It is likely that a bug in one will be a bug in another, but it is also true that some bugs in one will never be of a concern to the other or will never appear because their running environment is significantly different. For example, we saw bugs in the Linux driver related to ACPI management that do not exist in FreeBSD.

We are currently evaluating, using tools such as CCFinder, code phenotypes and evolution across different Unix kernels, *e.g.*, OpenBSD, FreeBSD, and Linux. They indeed contain several thousands of file pairs with clones, hundreds of them having over 30% code possibly cloned from one side to the other. In many cases, the change history of these files is substantially different in each system.

Like Rachel and Roy, clones phenotypes share a significant portion of a common clone genotype that determine their main features (and defects). As time progresses, like replicants, each clone will develop its own identity with its own unique features.

References

[1] M. W. Godfrey and D. M. German. The Past, Present, and Future of Software Evolution. In *Frontiers of Software Maintenance, 2008 (FoSM 2008)*, pages 129–138, Sept. 2008.