Supporting Information

Yan et al. 10.1073/pnas.0914771107

SI Text

Extraction of Call Graphs from the Linux Kernel. We downloaded 24 stable versions of kernel from the Linux Kernel Archives (http://www.kernel.org/), namely, 2.6.4 to 2.6.27, spanning from March, 2004 to October, 2008. In general, the release of a new version, say, from 2.5 to 2.6, is accompanied by major changes. We worked on the 24 releases restricted in version 2.6 and focused on the gradual evolution exhibited in these releases. Nevertheless, kernels 2.6.1 to 2.6.3 failed to compile in our system. The source codes were compiled on a MacBook with a 2 GHz Intel Core 2 Duo processor and 2 GB of memory by using the compiler GCC 3.4.6. As a result of this particular hardware architecture, additional patches were required for some of the versions in order to compile. More specifically, the patch file linux-2.6-seg-5.patch was patched to kernels from v2.6.4 to v2.6.12. The call graphs of the compiled code were extracted by using the tool CodeViz (release 1.0.11). CodeViz is a static call graph generator that works for C and C + + programs. We used the script genfull in CodeViz, with an option –g cobjdump. Under this setting, CodeViz analyzes a binary file instead of the source code. Naming collision is therefore unlikely. If a function is called multiple times during execution, the call graph does not take into account the number of callings. The resultant network is unweighted. By using the parameters described, CodeViz extracted the system-level functions without inline functions and macros, and thus we found no recursive function in the final network—i.e., no self-loop.

Persistent Functions and the Rate of Evolution. A function from call graph v2.6.27 is defined as persistent if it appears in all the earlier compiled call graphs (v. 2.6.4 to v. 2.6.26). The rate of evolution was quantified by the frequency of revision, which was found by searching for whether the persistent functions appear in the patch files: patch-2.6.4 to patch-2.6.27 (ftp://ftp.kernel.org/pub/linux/kernel/v2.6/). In this definition, we did not take into account the precise changes in the code of the subroutine. A function was regarded as revised if there was any change in its code.

Supplementary File 1. *The call graph of the Linux kernel (v2.6.27).* http://networks.gersteinlab.org/callgraph

Supplementary File 2. *The persistence of functions in the Linux kernel (v2.6.27).* http://networks.gersteinlab.org/callgraph