SAND2007-3436P

Self-assembling Software Promises Relief to Frustrated Software Users

A revolutionary self-assembling software (SAS) technology that will allow end-users to easily use, inspect, and directly modify software capabilities, is under development at Sandia National Laboratories. The concept, a biologically-inspired approach to software development and use, is currently the subject of eight U.S. patent applications, one of which has been granted.

The inventors, Ann Bouchard and Gordon Osbourn, call SAS "a transparent and safe form of lightweight programming that users of any expertise level can carry out." They expect that their new technology will allow users to take on tasks that are currently possible only by hiring custom programmers or by trying to cherry-pick the few simple tools among vast numbers of features offered in any given application. It will also allow programmers to focus their expertise on more difficult programming tasks.

Software has become a huge and essential component of the world economy. However, it presents some serious shortcomings that Bouchard and Osbourn are addressing through SAS. Software does not scale up well to large sizes and requires hand-crafting by skilled programmers. Software systems are also difficult to learn and often frustrating to use by all but the most highly-trained experts. Additionally, software products often require extensive learning periods to carry out even simple tasks. Most importantly, software products do not allow the user to adapt their capabilities to directly address the user's specific needs.

"Abandoned software projects and bug-fixes cost the US an estimated $60-90 billion every year, and "bugs" are essentially impossible to remove from large software systems," says Osbourn. "The only significant method for reducing software development costs is being implemented on a large scale today – replace expensive US software engineers with less expensive foreign software engineers.

"The SAS software self-assembles new functionality while it is running," he explains. "The goal of SAS is to indirectly free programmers from many tedious and custom tasks, such as GUI (graphical user interface) programming, and from the need to frequently modify software code under development due to changing requirements from customers."

Bouchard and Osbourn recently demonstrated the software functionality of SAS self-assembling pathways by implementing an environment with an adaptable GUI. The GUI provides an intuitive interface that allows users without programming experience to safely guide the self-assembly processes to achieve simple but custom functionality as needed. The GUI contains familiar artifacts (books in bookcases, context-sensitive tool menus), as well as a powerful new capability that the user can use to monitor and control all actions in the environment. Work in progress will provide the environment with a "search and rescue" capability that will allow the user to inspect any past actions that affect any kind of data (in various states of completion), and selectively undo, re-create, or replace some or all of those past actions and data. Bouchard and Osbourn expect to pilot the SAS environment inside Sandia next year.

Learning from biology

SAS is based on the way in which the protein networks in living cells interact to carry out complex information processing tasks. It transfers a model of robust protein self-assembly into the virtual world, employing a new kind of software entity termed a virtual protein "machine."

Protein networks carry out many of the actions needed to grow and stay alive. Their minute "construction crews" carry out complex programs that assemble, tear down, transport, and repair molecular structures within the cell, as well as detecting and responding to external stimuli such as food and threats. These structures are easily altered, temporarily combined, and even disassembled, in response to changing conditions or signals. Thus, pathways for information and material flow can be created "on-the-fly" as needed and dismantled when no longer needed.

While these single-cell capabilities seem simple compared to those of complex organisms such as humans, they outshine the capabilities of even the most complex man-made systems. But like them, SAS machines have dynamic binding sites that can selectively bind to sites of other machines. "The binding or unbinding of sites can trigger data passing, code execution, or the exposing, hiding or altering of other sites," explains Osbourn. "Changes in other sites can in turn trigger subsequent binding or unbinding of those sites, resulting in further data passing, execution, etc.

"The net result is the dynamic formation of functional software assemblies (a collection of bound machines) that provide data and execution pathways. The software functionality at any given time is determined by the current structure of the machine assemblies. This in turn is determined by the self-assembly and disassembly processes. A pathway can be reconfigured (machines unbound, and possibly re-bound to other machines), and this reconfiguration (effectively, assembly of a different pathway) results in different functionality."

Sandia technical contact: Gordon Osbourn, gcosbou@sandia.gov, (505) 844-8850
Sandia media contact: Nigel Hey, nigel@swcp.com, (505) 898-6679

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000