ParaSail: Less Is More when Parallel Programming for Multicore

As part of the merger between SoftCheck and AdaCore, the programming language ParaSail will now be under AdaCore auspices. ParaSail (Parallel Specification and Implementation Language) was specifically designed to parallelize programming parallel and productive, in contrast with the experience one gets from most existing programming languages. ParaSail achieves its goal by removing impediments that stand in the way of utilizing multiprocessor chips, and by adding an inherently scalable, passively parallelizing programming model. The impediments are divided into two broad categories: the introduction of data races, and deadlocks, as well as the other conformance issues.

Despite the significant simplification, ParaSail still provides a familiar class-and-interface-based object-oriented programming model. Programming constructs result from the following three distinctive features: realistic concurrency, explicit exception handling, and on-the-fly compilation.

The key point is that although this function does not appear to be explicitly parallel, the ParaSail semantics imply that the two recursive calls of $Word\_Count$ are in fact executed in parallel. Thus recursive calls then cause the "divide-and-conquer" approach to counting the words in the string. Thus $N$ threads are spawned for a string of length $N$, with each thread added together in a computation tree of depth $O(N)$. A potential speedup of $O(N)$ relative to a sequential algorithm is achieved.

For more information about ParaSail, please visit the web sites www.adacore.com or www.parasailprogramming-language.blogspot.com.

AdaCore and SofCheck Join Forces

AdaCore has merged with SoftCheck, Inc., a Lexington, Massachusetts-based company that offers an advanced, full-featured software quality (ASQ) company that has specialized in static analysis technology and Ada language expertise. SofCheck engineers Sheikh Bernsten, Mike Cleaves, Minelle Gari, and Mahau Roy have joined the AdaCore technical team, and SofCheck founder and Chairman Tucker Taft has been named as AdaCore’s Director of Language Research. This new role will be in charge of efforts to advance the state of the art in language design and implementation for high-reliability, safety-critical, and high-security systems.

With this merger AdaCore has acquired SoftCheck’s products, including static error detection tools. AdaCore has also acquired ParaSail, a new Parallel Specification and Implementation Language designed by Mr. Taft and intended for high-reliability applications on multiprocessor platform technologies.

When adding high-reliability applications, ParaSail personnel share a long and direct involvement in the design of the Ada programming language, and the two companies have specialized in software development tools that nicely complement each other. AdaCore’s advanced static analysis technology fits in smoothly with the GNAT Pro development environment, and prior to the merger the two companies jointly developed AdaCore’s CodePeer product, an efficient and accurate code reviewer that incorporates the SoftCheck static analysis engine. The merger will facilitate further enhancements to CodePeer, provide personnel and technology resources for new product offerings, and also simplify support arrangements for CodePeer customers.

For up to date information on conferences when AdaCore is participating, please visit www.adacore.com/category/press-center/events/
In addition to its Ada 2012 support, the GNAT Pro compiler exhibits a complete unit testing infrastructure for complex projects. Other tool enhancements include new and improved error messages, and performs optimizations for array handling of controlled types, provides new information, please visit the AdaCore web site. GNAT Pro insider Spring/Summer 2012

Inspired by challenges they observed while collaborating with Rockwell Collins engineers on certified embedded information assurance applications, and with US Food and Drug Administration engineers on next-generation medical devices, SanToS researchers are working on several innovative enhancements to SPARK’s annotations for software contracts:

- Precise information flow specification and checking. Embedded information assurance components such as crytographic controllers and cross-domain systems often have information flow policies that are conditional (i.e., data can flow from one domain to another only under certain conditions). This requires specifying how information flows through individual cells or ranges of elements in complex data structures built using arrays and records. SanToS researchers are extending SPARK’s data annotations to support declaration of conditional flows and precise descriptions of flows through arrays.

- Highly automated checking of complex pre/postconditions. Checking complex pre/postconditions can require significant manual interactions with SPARK’s Pre/Postcheck. SanToS researchers have developed an alternate tool chain based on symbolic execution that can perform bounded checking of such contracts in a completely automated fashion, and can automatically generate unit test cases representing the paths explored in the analysis. SanToS researchers are currently working with AdaCore and Altran Praxis to include some of these functionalities in the upcoming SPARK 2012 framework.
In addition to its Ada 2012 support, the GNAT Pro compiler exhibits improvements in memory management and a more efficient handling of controlled types, providing new warning on infinite loops, and improved handling of unused assignments. This major new release supports the upcoming Ada 2012 language revision, the full run-time library and the full file system. SPARK in Formal Verification Research

< current releases >

GNAT Pro 7.0
This major new release supports the upcoming Ada 2012 language revision, introduces a flexible and powerful testing tool (GNAITester), incorporates a range of improvements (many based on user suggestions), and adds several new platforms. Some of the key enhancements:

- Compiler: adds support for all new features of the Ada 2012 language revision;
- Improved analysis of range conditions;
- Detection of access to global uninitialized variables;
- New functional tests in GNAITest and GNATMetric mentioned above, since these are incorporated in CodePeer.

CodePeer 2.1
CodePeer 2.1, the latest version of AdaCore's advanced static analysis / abstract interpretation tool, introduces a number of significant enhancements, including:

- Complete support for all new features of the Ada 2012 language revision;
- Improved analysis of range conditions;
- Detection of access to global uninitialized variables.

GDB in GNAT Pro 7.0
GDB’s scripting capabilities have been significantly enhanced by the integration of Python. Python has been used successfully in other AdaCore tools (such as GPS) as a powerful and flexible extensibility and customization mechanism. For an overview of how to use this facility, please refer to AdaGems #112 and #122 at www.adacore.com/adaswers/gems/.

< in the pipeline >

GNAT Pro for Wind River Linux
GNAT Pro will be available during Q3 2013 on all Wind River host platforms for Wind River embedded Linux, versions 4.2 and 4.3, for PowerPC and PowerQUICC III (e500v2 family) processes. This will be a full version of GNAT Pro with support for GPS and a full GNATIntegration with Wind River Workbench.

GNATcoverage and GNATEmulator

GNATcoverage, a non-invasive source and object coverage tool, and GNATEmulator, a machine emulator and virtualizer based on qemu technology, recently added support for several new platforms: PowerPC ELF, Leon ELF, and P55xx/E500V2 ELF for GNAITest Coverage, and VxWorks 6 for GNATEmulator. During Q3 2012 the planned enhancements include improvement of the GPU plugins for both products, and GNAITestCoverage support (except for M6090) of VxWorks 6 with the full run-time library.

GtkAda
GtkAda 3.0 has come out later this year, will support version 3 of the Gtk library. This will bring a number of enhancements including a simplified API, several new widgets, complete new features such as CSS-based theming, and an experimental HTML5 back-end. Gkada 3 will allow new features and code sharing, and a new generation of controls in the upcoming version, such as support for low-level drawing with the Cairo graphics library, printing support, and the Gdade GUI builder.

New Course on DO-178C
Following its participation in the group that developed the new DO-178C, avionics software certification standard, AdaCore is now offering a four-day course for engineers and project managers. The course covers the key concepts behind DO-178C, including a summary of the differences with DO-178B, shows how to apply the guidance in the Object-Oriented Technologies and Related Techniques supplement (DO-332), and explains how to use the Ada-language and GNAT Pro tool suite to comply with the certification objectives. For further information, please contact info@adacore.com.

< academia corner >

Interview with Nicolas Setton

Product Manager, GPS and GtkaDa, AdaCore EU

Tell us about your background and how you came to be involved with Ada and AdaCore. What is your current role?

My initial encounter with Ada was as an undergraduate at Télécom ParisTech (ENS). I can remember the shock when the compiler rejected my program because it refused to perform an implicit conversion between pointer (I mean access) types. At first I felt quite outraged (“What? The C compiler never complained like this!”). After some effort, I understood that I was in fact dealing with a very powerful tool—and the fun began! Later, my Computer Science professor told me about an internship opportunity at AdaCore, and I joined the company’s Paris office right after graduating.

I’ve worked on a variety of projects, mainly on the GPS IDE, and on our infrastructure for quality management and tool qualification. I also took charge of some interesting business management projects, like setting up the virtualization cluster. I am currently Product Manager for the GtkaDa Graphical toolkit and GPS, which entails what you would expect: coordinating with the technical team, developing product roadmaps, etc.

Based on your experience with graphical software and workstation technology, what do you see as future trends for GUis?

Many GUI decisions are simply good adaptations to hardware changes, so, by looking at the hardware of today and the directions it is taking, we can make some predictions about the GUis of tomorrow. As an example, the geometry of our workstation screens has evolved to an ever-increasing width: at present, 1366x768 has topped 1024x768 as the most widely used resolution. Earlier this year, I noticed that the layout of my email program changed from an up-down division to a side-by-side approach. This is not a coincidence: the GUI has been cleverly manipulated to match the evolution of the hardware. I expect to see more desktop interfaces adapting to this wider form factor, for example by removing controls from the top or bottom and putting them on the sides. The pixel pitch of our displays will reach a maximum: some devices can already match the capabilities of the human retina, and we will never need to exceed this capacity. GUI engineers will have to find new ways of innovating.

With regard to input devices, things are even more interesting! After 20-odd years of very few improvements (the mouse wheel being the only one to come to mind that had an influence on GUI design), a breakthrough has occurred with the advent of multi-touch surfaces and displays. I think this is a very promising breeding ground for new paradigms! One example is model-driven engineering: I would love to manipulate UML diagrams directly on a tablet, while I edit code on my current workstation. In general, the ability to “touch” our data I think a game-changer, and I look forward to seeing this integrated with the code-quality dashboards and other reporting tools that are emerging.

The industry of today is more demanding of aesthetically pleasing interfaces than it was 20 years ago, involving graphical designers and sometimes ergonomists. The toolkits help, by separating the GUI design from the actual coding. The latest generation of toolkits allows users without programming skills to create GUI layouts, and separates theming and customizing activities from the GUI development through CSS-like approaches.

At some point I would love to see on-eye displays operated by micro-gestures or eye movements, but this might just be my love for science fiction.

Any hobbies or outside interests that you’d like to share?

Aside from programming or assembling computers? I was recently both invigorated and traumatized by a brush with amateur theatre. I now have a very deep respect for people who are able to put themselves on stage and perform. I’m sure I’ll have another go in the future!

< focus >

Open-DO Update

The Open-DO initiative, which seeks to bring the benefits of open source and agile technology to the safety certification community, continues to grow. The ERTS 2012 conference in Toulouse this past February featured an Open-DO session with presentations on topics such as the integration of formal methods with testing; all the papers from this session are available at www.erts2012.org/Default.aspx?Id=1050&id=112907A. Several of the projects which are part of the Open-DO initiative are Riposte, the Nose Gear Challenge Problem, and explanatory documents on DO-178C. They can be found at forge.open-do.org.

An Open-DO day will be organized in early 2013 where these projects, and others, will be presented. Details will be posted on the Open-DO web site www.open-do.org later this year.

Webinar Schedule

AdaCore’s 2012 Insight Webinar series has covered SPARK Pro 10.1, the new GNATTest Unit Test Harness Generator tool, and CodePeer 2.1. To view any of these, or to learn more about future webinars, please visit www.adacore.com/knowledge/webinars.
As part of the merger between SoftCheck and AdaCore, the programming language ParaSail will become part of AdaCore. ParaSail is a parallel specification and implementation language designed specifically for parallel programming in Ada. ParaSail is designed to achieve parallelism in a manner that is not only easier for developers to understand and maintain, but also more efficient and scalable. ParaSail achieves its goal by removing impediments that stand in the way of using multicore chips, and by adding an inherently safer, pervasive parallel programming model.

ParaSail programs are written in a straightforward manner, with a focus on readability and maintainability. The language is designed to be easy to learn and use, and it supports a wide range of parallel programming paradigms. ParaSail also includes a set of built-in parallel constructs, such as parallel loops and parallel assignments, as well as a set of parallel control structures, such as parallel if-else statements and parallel case statements.

One key point is that although this function does not appear to be explicitly parallel, the AdaCore semantics imply that the two recursive calls of $\text{Word_Count}$, in the initialization of $\text{S}$ are fact evaluated in parallel. Those recursive calls then cause the “divide and conquer” approach to counting the words in the string. Thus X ticks are spared, as a position $\lfloor \frac{N}{2} \rfloor$ with then results added together in a computation tree depth $\log_2(N)$. A potential speedup of $\log_2(N)$ relative to a sequential algorithm is achieved.

ParaSail is designed to be easy to learn and use, and it supports a wide range of parallel programming paradigms. ParaSail also includes a set of built-in parallel constructs, such as parallel loops and parallel assignments, as well as a set of parallel control structures, such as parallel if-else statements and parallel case statements.

The design process for AdaCore 2012 is complete. The Ada Rapporteur Group (ARG) has submitted the reference manual for this latest version of the language to WSG. The working group in charge of Ada in charge of Ada. After the document has been approved by the WSG national delegations, the International Organization for Standardization (ISO) is expected to approve the new standard before the end of 2012. The new Ada 2012 reference manual is available at www.ada-auth.org/standards/ada12.html.

AdaCore and SofCheck Join Forces

AdaCore has merged with SofCheck, Inc., a Lexington, Massachusetts-based open-source software quality (ASQ) company that has specialized in static analysis technology and Ada language expertise. SofCheck engineers Mr. Taft has been named AdaCore’s Director of Language Research, and Mr. Taft will be in charge of efforts to advance the state of the art in language design and implementation for high-reliability, safety-critical, and high-security systems.

With this merger AdaCore has acquired SofCheck’s products, including static error detection tools. AdaCore has also acquired Parasail, a new Parallel Specification and Implementation Language designed by Mr. Taft.

For more information about ParaSail, please visit the website parasail-programming-language.blogspot.com/ or www.adacore.com/labs/.