AdaCore Vendor Presentation

Ada-Europe 2018
Lisbon, Portugal
June 20 2018

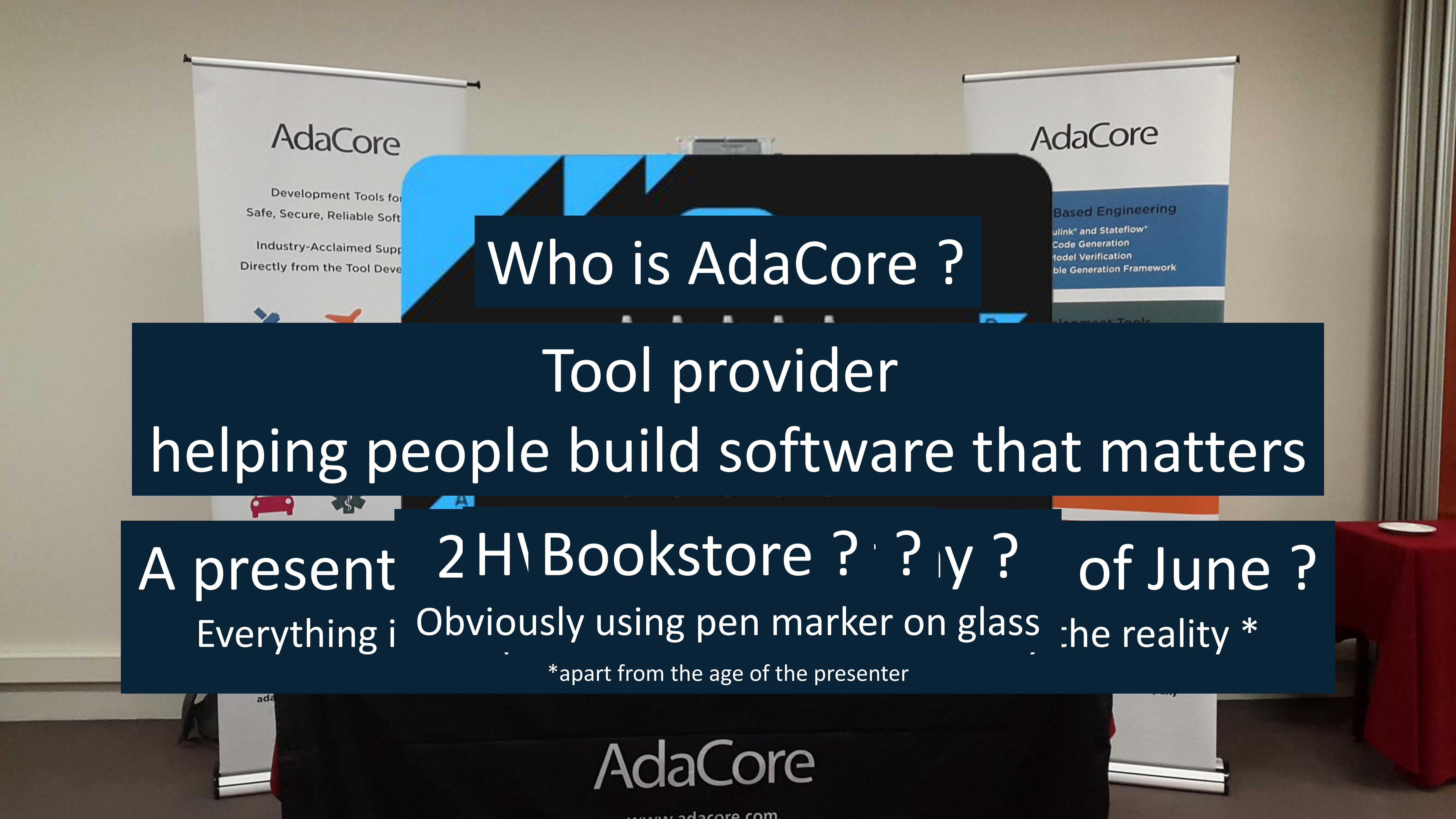
Eric Perlade, Technical Account Manager
perlade@adacore.com

Outline

- Who is AdaCore?
- Major products
- GNAT Pro product lines
- GNAT Pro 18
- Safety certification
- AdaCore for the community

AdaCore

Who is AdaCore?



Who is AdaCore ?

Tool provider
helping people build software that matters

A present 2 H\ Bookstore ? ' ? ly ? of June ?

Everything i Obviously using pen marker on glass the reality *

*apart from the age of the presenter

Who We Are

Founded in 1994 (US) and 1996 (France) to commercialize / productize the GNAT Ada technology

- Software development tools/environments for organizations building critical systems
- Headquarters in New York City and Paris
- Staff comprises around 100 people, distributed worldwide

Deep technical expertise in many areas

- Ada Language
- Compiler Technology
- Embedded Computing
- Static Analysis
- Dynamic Analysis
- Formal Methods
- Safety Certification
- Model-based development

Who We Are

Customers worldwide in the most demanding domains

- Commercial and military avionics
- Air Traffic Management / Control
- Space
- Rail
- Medical
- Financial services

www.adacore.com/customers

We Are Dedicated to…

Ada and SPARK languages

- Complete implementation of latest standards (Ada 2012, SPARK 2014)
- GNAT Academic Program for colleges and universities

Active participation in professional organizations

- ISO WG9 Ada Rapporteur Group
- Ada-Europe, SIGAda, Ada Resource Association, …
- FACE Consortium - `www.opengroup.org/face`

Improving High-Integrity software development

- RTCA/EUROCAE SC205/WG71 (DO-178C/ED-12C Committee)
- Booklets on using AdaCore technologies with rail (EN 50128) and airborne software (DO-178C) standard

Open collaboration

- Freely-Licensed Open Source Software (“FLOSS”)
- Many projects on GitHub

Major Products

AdaCore Products

GNATPro

- GCC-based compilation system, IDEs, toolset, libraries
- Multi-language support: Ada (all versions), C, …
- Native and cross platforms
  - Targets include RTOS (e.g. Wind River VxWorks, SYSGO PikeOS, LST LynxOS) and bare metal

CodePeer

- Advanced static analysis tool for Ada
- Detects potential bugs / vulnerabilities including many from CWE

AdaCore Products

SPARK Pro

- Formal analysis tool for proving program properties (SPARK 2014)
- SPARK Adoption Guidance booklet (with Thales)

QGen

- Model-based development environment with debugger and qualifiable code generator to SPARK or MISRA-C
- Processes a safe subset of Simulink® and Stateflow® models

All products backed by “front-line support”

GNAT Pro product lines

GNAT Pro product lines

GNAT PRO ASSURANCE

Designed for certification and long-lived projects

GNAT PRO ENTERPRISE

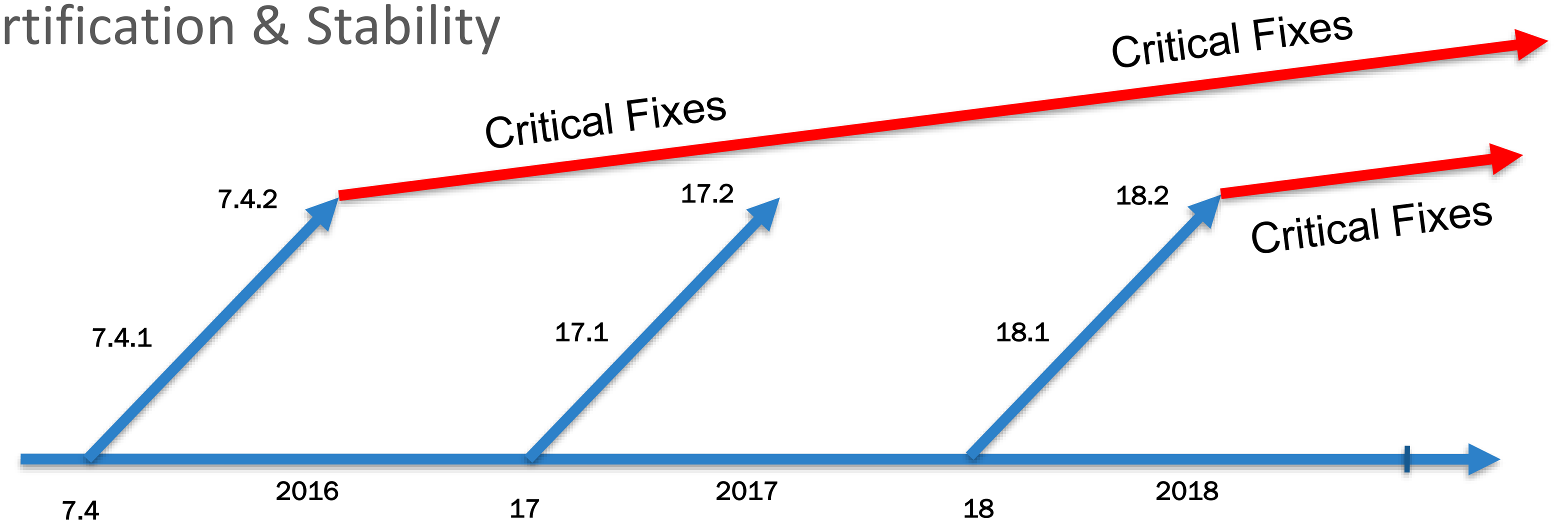
Industrial-Grade Software Development

GNAT PRO DEVELOPER

Getting up and running with Ada

GNAT Pro Assurance

For Certification & Stability



- Sustain branches will provide
  - Safety analysis on known-problems (including impact analysis)
  - Potential safety-critical fixes
  - Access to fixed releases beyond the yearly branch cut

GNAT Pro 18

GNAT Pro 18 overview

- 57 platforms (52 cross, 5 native)
- Support for Ada 83, 95, 2005, 2012, SPARK 2014
  - Including latest Ada 2012 corrigendum
- Based on GCC 6.4 & GDB 7.10
- SPARK Discovery available to all GNAT Pro customers
- GNATstack available to all Enterprise and Assurance customers

New ports

- PowerPC 64bits VxWorks 7
- x86 32bits VxWorks 7
- Aarch64 (ARM 64bits) bare metal

GNAT – Support for Ada 202x

- **pragma Ada_2020**
- **@ shorthand**

```
Very_Long_Expression (Very_Long_Variable).Very_Long_Field := @ + 1;
```

- **delta aggregates**

```
Post => X = (X’Old with delta Foo => 12, Bar | Baz => 42);
```

```
   type Powers is array (1..5) of Integer;
   type Table is array (1..4) of Powers;
   Thing : Table;
begin
   Thing := (others => (for I in Powers'range => I));
   Thing (2) := (@ with delta for all J in 3..5 => @ (J) ** 2);
```

GNAT – Improved Code Generation

- Automatic reordering of components in record types with discriminants
- New pragma No_Component_Reordering
- Improved inlining of subprograms coming from a parent package
- Various improvements in C binding generation
  - More accurate binding for enumerals and constants, supporting the declaration of const-qualified variables.
- Various other performance enhancements

GNAT – Warning Improvements

- Option to treat warnings about run-time exceptions as errors (-gnatwE)

- New warning for ineffective use and use type clauses (-gnatwu)

GNAT - Cross Platforms

- Avoid -mlongcall code size bloat
  - LynxOS-178 2.2.4
  - VxWorks 6/7 RTP
- Shared libgnat available on all cross Linux targets
- Pre-built XMLAda on all cross targets with a full runtime
- Native ZFP runtime comes preinstalled

GNAT - Bare Metal

- GNAT Pro C for PowerPC and p55
- SMP support for ARM Cortex-A
- Clean BSP / Runtime separation
- Support arbitrary run-time additions
- Use ARM semihosting for Ada.Text_IO (arm-elf)
- Improvements in GPS integration

GNAT Pro for C and C++

- C support is now available on:
  - all Natives
  - all VxWorks 6 configurations
  - all VxWorks 7 configurations
  - all Bare Metal targets
  - all Embedded Linux configurations
- C++ support is available on all natives

- Features include:
  - GNATstack
  - latest GCC, C and C++ versions
  - compatibility with GNATcoverage
  - GNATemulator (when available)
  - availability of Assurance product
  - GPS support
  - GNAT Tracker support
  - Ada/C++ OOP & exception compatibility
  - endianness specification clauses

Tools

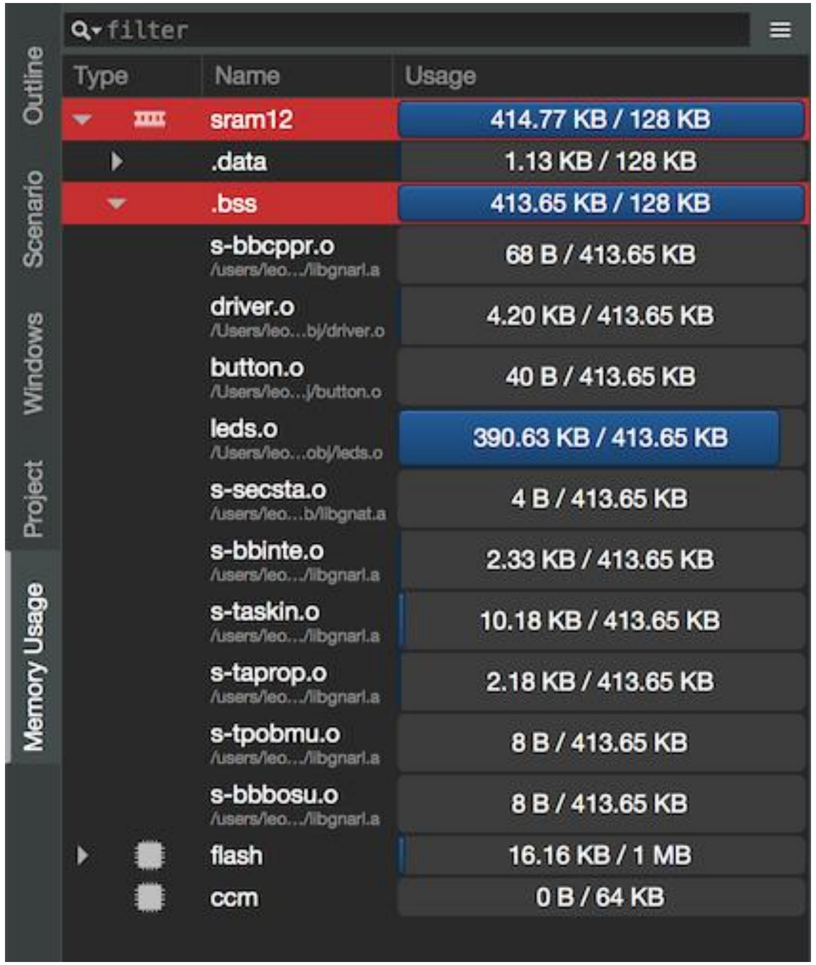
- GNATcheck – Qualified Coding Standard Checker
  - 24 new rules implemented and qualified
  - Several qualification kits delivered
- GNATcoverage – Qualified Coverage
  - Qualification material migrated to DO-178C & Ada2012
  - Support for probes iSystem and Lauterbach
  - Support for compressed ELF sections
  - Work in progress: instrumentation based scheme
- GNATtest – Unit Testing Framework
  - More options to control which parts of the user application to select for testing and for stubbing
  - More options to customize the generated harness

GPR2

- New GPR engine using langkit/libadalang
- Ability to work with multiple project hierarchies
- Published on GitHub: github.com/AdaCore

GPS – UI Improvements

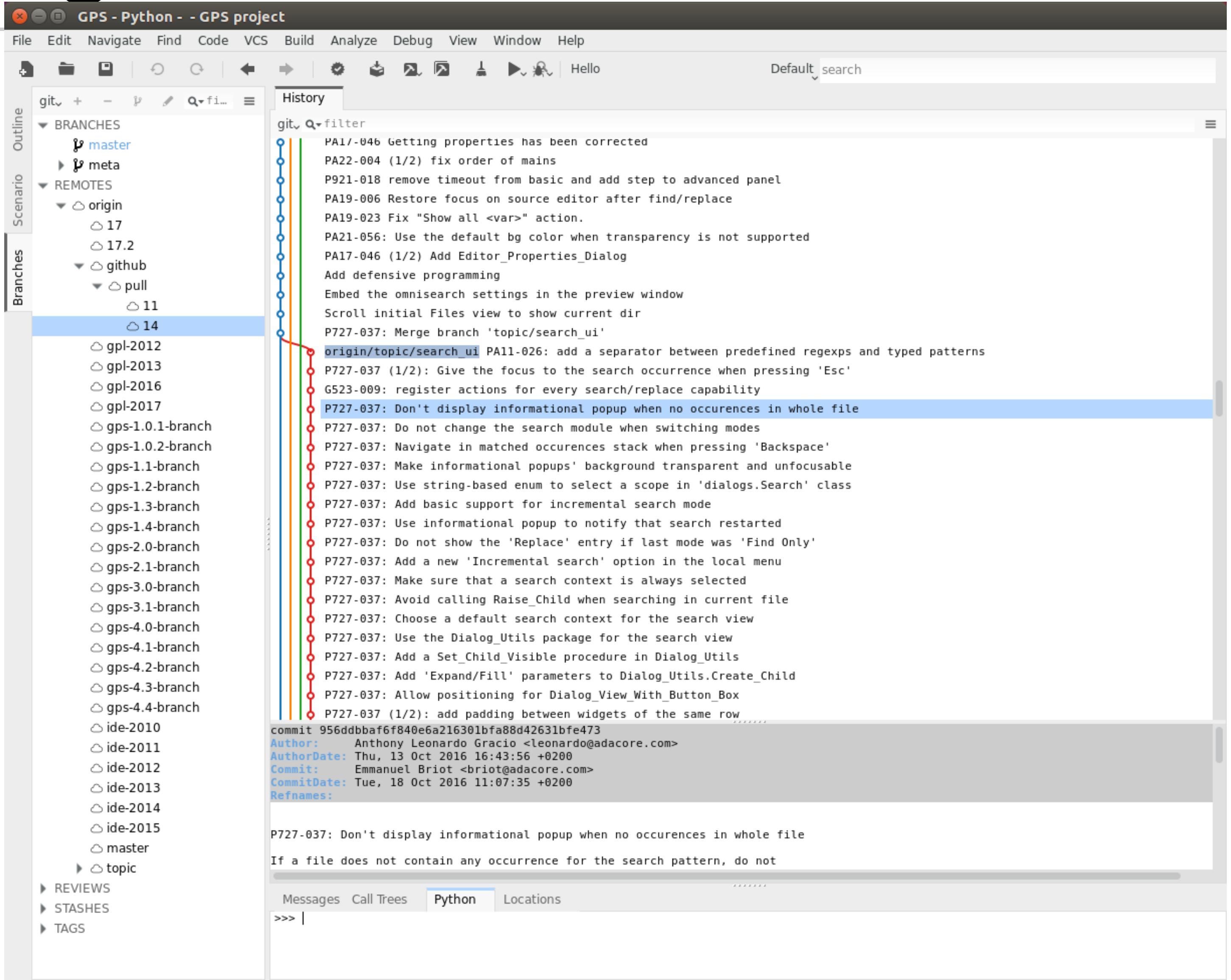
- Improved experience for newcomers, including a revamped welcome screen, and a "first time" assistant
- New, clearer Search dialog
- New Memory Usage View
- Build & Run, Build & Debug buttons
- Button to switch between perspectives



Type	Name	Usage
	sram12	414.77 KB / 128 KB
	.data	1.13 KB / 128 KB
	.bss	413.65 KB / 128 KB
	s-bbcppr.o /users/leo…/libgnarl.a	68 B / 413.65 KB
	driver.o /Users/leo…bj/driver.o	4.20 KB / 413.65 KB
	button.o /Users/leo…j/button.o	40 B / 413.65 KB
	leds.o /Users/leo…obj/leds.o	390.63 KB / 413.65 KB
	s-secsta.o /users/leo…b/libgnat.a	4 B / 413.65 KB
	s-bbinte.o /users/leo…/libgnarl.a	2.33 KB / 413.65 KB
	s-taskin.o /users/leo…/libgnarl.a	10.18 KB / 413.65 KB
	s-taprop.o /users/leo…/libgnarl.a	2.18 KB / 413.65 KB
	s-tpobmu.o /users/leo…/libgnarl.a	8 B / 413.65 KB
	s-bbbosu.o /users/leo…/libgnarl.a	8 B / 413.65 KB
	flash	16.16 KB / 1 MB
	ccm	0 B / 64 KB

GPS – New VCS Engine

- Simpler to extend and customize
- First-class support for Git workflows, including a history view



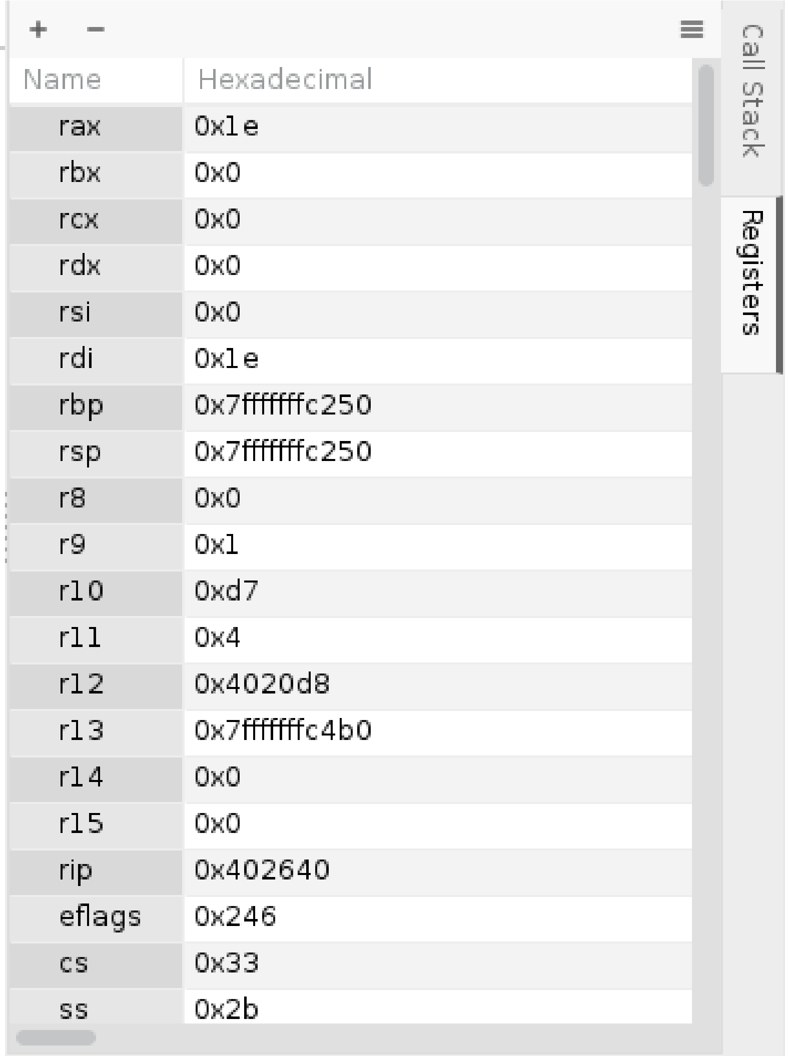
```
PA17-046 Getting properties has been corrected
PA22-004 (1/2) fix order of mains
P921-018 remove timeout from basic and add step to advanced panel
PA19-006 Restore focus on source editor after find/replace
PA19-023 Fix "Show all <var>" action.
PA21-056: Use the default bg color when transparency is not supported
PA17-046 (1/2) Add Editor_Properties_Dialog
Add defensive programming
Embed the omnisearch settings in the preview window
Scroll initial Files view to show current dir
P727-037: Merge branch 'topic/search_ui'
origin/topic/search_ui PA11-026: add a separator between predefined regexps and typed patterns
P727-037 (1/2): Give the focus to the search occurrence when pressing 'Esc'
G523-009: register actions for every search/replace capability
P727-037: Don't display informational popup when no occurences in whole file
P727-037: Do not change the search module when switching modes
P727-037: Navigate in matched occurences stack when pressing 'Backspace'
P727-037: Make informational popups' background transparent and unfocusable
P727-037: Use string-based enum to select a scope in 'dialogs.Search' class
P727-037: Add basic support for incremental search mode
P727-037: Use informational popup to notify that search restarted
P727-037: Do not show the 'Replace' entry if last mode was 'Find Only'
P727-037: Add a new 'Incremental search' option in the local menu
P727-037: Make sure that a search context is always selected
P727-037: Avoid calling Raise_Child when searching in current file
P727-037: Choose a default search context for the search view
P727-037: Use the Dialog_Utils package for the search view
P727-037: Add a Set_Child_Visible procedure in Dialog_Utils
P727-037: Add 'Expand/Fill' parameters to Dialog_Utils.Create_Child
P727-037: Allow positioning for Dialog_View_With_Button_Box
P727-037 (1/2): add padding between widgets of the same row
```

```
commit 956ddbbaf6f840e6a216301bfa88d42631bfe473
Author:     Anthony Leonardo Gracio <leonardo@adacore.com>
AuthorDate: Thu, 13 Oct 2016 16:43:56 +0200
Commit:     Emmanuel Briot <briot@adacore.com>
CommitDate: Tue, 18 Oct 2016 11:07:35 +0200
Refnames:


P727-037: Don't display informational popup when no occurences in whole file

If a file does not contain any occurrence for the search pattern, do not
```

```
>>> |
```

GPS – Debugger Support

- Support for interfacing to the debugger via GDB-MI
- New Registers view, allowing registers edition
- Improved control on representation of debugged data



Name	Hexadecimal
rax	0x1e
rbx	0x0
rcx	0x0
rdx	0x0
rsi	0x0
rdi	0x1e
rbp	0x7fffffffc250
rsp	0x7fffffffc250
r8	0x0
r9	0x1
r10	0xd7
r11	0x4
r12	0x4020d8
r13	0x7fffffffc4b0
r14	0x0
r15	0x0
rip	0x402640
eflags	0x246
cs	0x33
ss	0x2b

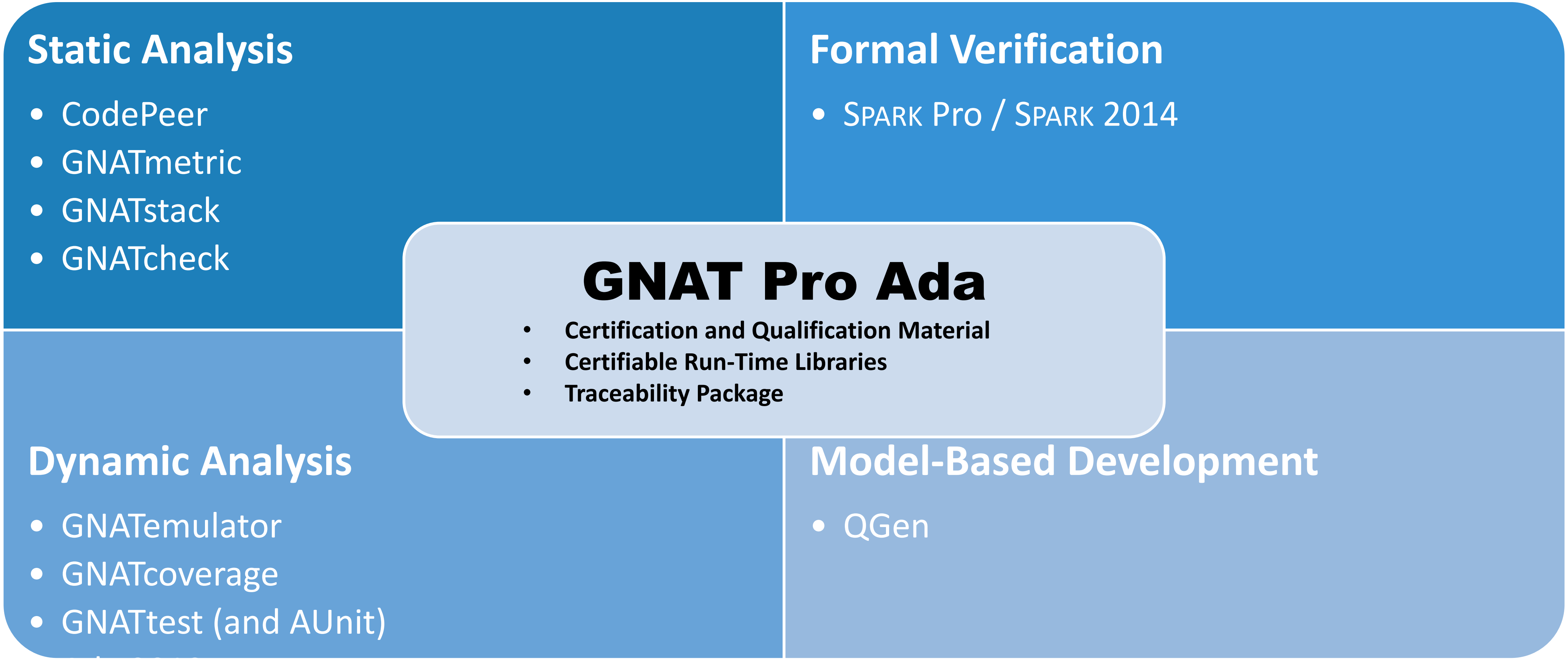
GNAT CCG

Common Code Generator

- GNAT Compiler that takes a subset of Ada as input and generates MISRA-C compliant code as output
- ZFP runtime: No tasking, no controlled types, no exception propagation
- Also removed from ZFP: no tagged types, only representation clauses that can be mapped easily to C representation clauses (e.g. bitfields)
- Packed arrays supported
- All runtime checks except overflow checks
- All user assertions (assert, pre/postconditions, …)

Software Certification

AdaCore and Safety- and Security-Critical Software



Static Analysis

- CodePeer
- GNATmetric
- GNATstack
- GNATcheck

Formal Verification

- SPARK Pro / SPARK 2014

GNAT Pro Ada

- Certification and Qualification Material
- Certifiable Run-Time Libraries
- Traceability Package

Dynamic Analysis

- GNATemulator
- GNATcoverage
- GNATtest (and AUnit)

Model-Based Development

- QGen

Certification / Tool Qualification Experience

	Avionics DO-178B/C	Railway EN 50128	Space ECSS-E-ST-40C ECSS-Q-ST-80C
Run-time libraries	Cert (PowerPC / VxWorks653)	Ravenscar Minimal (PowerPC / bareboard)	Ravenscar SFP (ERC32, LEON2, LEON3 bareboard)
Tools	• Coding Standard Checker (GNATcheck) • Code Coverage (GNATcoverage) • Run-Time Error Detection (CodePeer)	• Compiler (GNAT Pro) • Coding Standard Checker (GNATcheck) • Code Metrics (GNATmetric) • Testing Framework (GNATtest& AUnit) • Code Coverage (GNATcoverage) • Absence of Run-Time Errors (GNATprove–SPARK Pro) • Data & Control Flow Analysis (CodePeer)	
Other	• Traceability Package		

AdaCore for the community

GNAT Community 2018

- BBC micro:bit first class support
- RISC-V support
- SPARK included in the package by default
- Arm-elf hosted on Mac

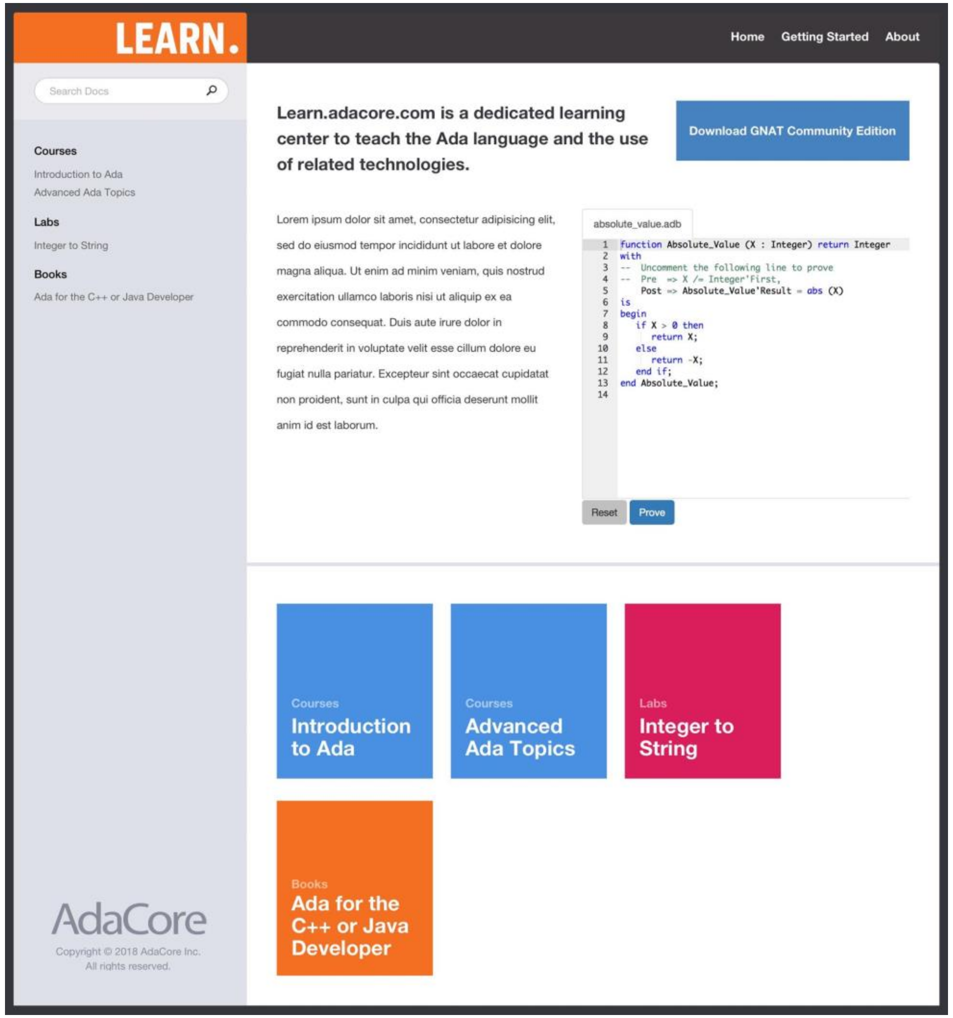
GNAT
COMMUNITY
2018

Make With Ada 2018

- Embedded software project competition sponsored by AdaCore.
- Open to individuals and small teams using the Ada or SPARK languages to develop dependable, open, inventive and collaborative projects.
- www.makewithada.org

Learn.adacore.com

- Interactive learning platform designed to teach the Ada and SPARK programming languages.
- Courses featuring hands-on labs and easy to understand code snippets,
- Opportunity to see, understand and experiment with the language capabilities.



Learn.adacore.com is a dedicated learning center to teach the Ada language and the use of related technologies.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

```ada
function Absolute_Value (X : Integer) return Integer
with
--  Uncomment the following line to prove
--  Pre  => X /= Integer'First,
    Post => Absolute_Value'Result = abs (X)
is
begin
   if X > 0 then
      return X;
   else
      return -X;
   end if;
end Absolute_Value;

```

Copyright © 2018 AdaCore Inc.
All rights reserved.

Thank you

AdaCore

European Headquarters

46 rue d'Amsterdam
75009 Paris
France

+33-1-4970-6716 (voice)
+33-1-4970-0552 (FAX)

North American Headquarters

150 W. 30th St., 16th Floor
New York, NY 10001
USA

+1-212-620-7300 (voice)
+1-212-807-0162 (FAX)

info@adacore.com www.adacore.com