

ADA USER JOURNAL

Volume 24
Number 1
March 2003

Contents

	<i>Page</i>
Editorial Policy for <i>Ada User Journal</i>	2
Editorial	3
News	4
Conference Calendar	42
Forthcoming Events	50
Articles	
Javier Miranda <i>“A Guided Tour to the GNAT Runtime”</i>	55
Alan Burns <i>“Increasing the Real-Time Expressive Power of Ada 95”</i>	60
Ada-Europe 2002 Sponsors	64
Ada-Europe Associate Members (National Ada Organizations)	Inside Back Cover

Editorial Policy for *Ada User Journal*

Publication

Ada User Journal – The Journal for the international Ada Community – is published by Ada-Europe. It appears four times a year, on the last days of March, June, September and December. Copy date is the first of the month of publication.

Aims

Ada User Journal aims to inform readers of developments in the Ada programming language and its use, general Ada-related software engineering issues and Ada-related activities in Europe and other parts of the world. The language of the journal is English.

Although the title of the Journal refers to the Ada language, any related topics are welcome. In particular papers in any of the areas related to reliable software technologies.

The Journal publishes the following types of material:

- Refereed original articles on technical matters concerning Ada and related topics.
- News and miscellany of interest to the Ada community.
- Reprints of articles published elsewhere that deserve a wider audience.
- Commentaries on matters relating to Ada and software engineering.
- Announcements and reports of conferences and workshops.
- Reviews of publications in the field of software engineering.
- Announcements regarding standards concerning Ada.

Further details on our approach to these are given below.

Original Papers

Manuscripts should be submitted in accordance with the submission guidelines (below).

All original technical contributions are submitted to refereeing by at least two people. Names of referees will be kept confidential, but their comments will be relayed to the authors at the discretion of the Editor.

The first named author will receive a complimentary copy of the issue of the Journal in which their paper appears.

By submitting a manuscript, authors grant Ada-Europe an unlimited license to publish (and, if appropriate, republish) it, if and when the article is accepted for publication. We do not require that authors assign copyright to the Journal.

Unless the authors state explicitly otherwise, submission of an article is taken to imply that it represents original, unpublished work, not under consideration for publication elsewhere.

News and Product Announcements

Ada User Journal is one of the ways in which people find out what is going on in the Ada community. Since not all of our readers have access to resources such as the World Wide Web and Usenet, or have enough time to search through the information that can be found in those resources, we reprint or report on items that may be of interest to them.

Reprinted Articles

While original material is our first priority, we are willing to reprint (with the permission of the copyright holder) material previously submitted elsewhere if it is appropriate to give it a wider audience. This includes papers published in North America that are not easily available in Europe.

We have a reciprocal approach in granting permission for other publications to reprint papers originally published in *Ada User Journal*.

Commentaries

We publish commentaries on Ada and software engineering topics. These may represent the views either of individuals or of organisations. Such articles can be of any length – inclusion is at the discretion of the Editor.

Opinions expressed within the *Ada User Journal* do not necessarily represent the views of the Editor, Ada-Europe or its directors.

Announcements and Reports

We are happy to publicise and report on events that may be of interest to our readers.

Reviews

Inclusion of any review in the Journal is at the discretion of the Editor.

A reviewer will be selected by the Editor to review any book or other publication sent to us. We are also prepared to print reviews submitted from elsewhere at the discretion of the Editor.

Submission Guidelines

All material for publication should be sent to the Editor, preferably in electronic format. The Editor will only accept typed manuscripts by prior arrangement.

Prospective authors are encouraged to contact the Editor by email to determine the best format for submission. Contact details can be found near the front of each edition. Example papers conforming to formatting requirements as well as some word processor templates are available from the editor. There is no limitation on the length of papers, though a paper longer than 10,000 words would be regarded as exceptional.

Editorial

Volume 24 of the Ada User Journal opens with an issue that features two highlights: our continued and close look into the ongoing Ada language revision process and an even bigger News section, which was afforded, for the expected gratification of the readership, by temporary changes in the Journal advertising policy.

After the introductory presentation of the language revision process that Pascal Leroy, the rapporteur of the ARG, the group that runs the revision process, offered to AUJ 23-4, we now turn our attention to one of major areas of activity in the revision process: the enhancement of the already powerful real-time programming model of the language, the intent of which is to further strengthen the leading role that Ada plays in the real-time and high-integrity application domains. Alan Burns, who represents the Ada real-time community in the ARG, gives us an overview of the current highlights of the real-time oriented proposals for language enhancements.

Finally, in this issue we have the pleasure of hosting a paper by Javier Miranda, which, after so often recurring the News section of the Journal, briefly walks us through his enormously informative work on the description of the internals of the GNAT runtime.

Let me close by reassuring the readers that we continue our effort to bring the Journal back to its calendar schedule, while being all the more eager to contributions from the Ada community at large.

*Tullio Vardanega
Padova
March 2003
Email: tullio.vardanega@math.unipd.it*

News

Dirk Craeynest (ed)

Offis nv/sa and K.U.Leuven. Email Dirk.Craeynest@offis.be

Contents

	page
Ada-related Organizations	4
Ada-related Events	4
Ada and Education	7
Ada-related Resources	9
Ada-related Tools	10
Ada-related Products	17
Ada and Linux	24
Ada and Microsoft	24
References to Publications	25
Ada Inside	27
Ada in Context	34

Ada-related Organizations

Tucker Taft at Helm of ARA and SofCheck

*From: Editorial Webmaster
<abrandon@sover.net>*

Date: Tue, 29 Oct 2002 11:56:04 -0500

Subject: [AdaIC] S. Tucker Taft at Helm of ARA and SofCheck

To: announce@adaic.com

S. Tucker Taft has rejoined the ARA as president of SofCheck. Tucker is a founder and president of both SofCheck, Inc., and the ARA. The ARA is seeing an old friend turn into the newest member of Ada software tool and compiler vendors.

Tucker is an old friend of the Ada community, too, as we need not tell most of you: From 1990-95, he led the Ada9X design team and succeeded in making Ada95 the first ISO standardized object-oriented programming language. In his twenty years with AverCom (AverStar, previously Intermetrics), Tucker became President of AverCom Corporation (a Titan Company).

For more information, please see the AdaIC site at

<http://www.adaic.org/news/sofcheck.html>

Editorial Webmaster,
webmaster@adaic.org

Join the Ada Resource Association

*From: Editorial Webmaster
<webmaster@adaic.com>*

Date: Tue, 04 Feb 2003 21:42:43 -0500

Subject: [AdaIC] Updates from the AdaIC News

To: announce@adaic.com

[...] the ARA has decided to open up membership from its original tool and compiler vendors to anyone interested in supporting Ada's future. [...]

New: Join the Ada Resource Association:
An introduction to the ARA's work for the Ada community and benefits of ARA membership.

<http://www.adaic.org/ARA/join.html> [...]

Ann Brandon, Editorial Webmaster

Ada-related Events

[The announcements reported below are a selection of the many Ada-related events organized by local groups. If you are organizing such an event, feel free to inform us as soon as possible. If you attended one please consider writing a small report for the Journal. -- dc]

Dec 6 - Public Technical Briefing on SPARK

From: rod@praxis-cs.co.uk (Rod Chapman)

Date: 27 Nov 2002 05:30:16 -0800

Subject: ANNOUNCE: Public technical briefing on SPARK, Houston

Newsgroups: comp.lang.ada

As a "warm up" event for ACM SIGAda 2002, Praxis Critical Systems will be presenting a public technical briefing on SPARK on Friday 6th December. This event is aimed at engineers and managers involved in the design and procurement of high-integrity software, and especially those involved in the aviation and space industries centred around NASA JSC.

Title: SPARK - A state-of-the-practice approach to High-Integrity Software.
Where: Houston Holiday Inn / NASA, 1300 NASA Road 1, Houston.

When: Friday 6th December 2002, 09:30 - 12:00.

RSVP: To sparkinfo@praxis-cs.co.uk [...]

Other contacts: rod.chapman@praxis-cs.co.uk www.sparkada.com

Abstract: Building truly high-integrity software requires a qualitatively different approach - simply "being more careful" is not good enough. In the UK and Europe, Praxis has pioneered an approach known as "Correctness by Construction" that has proven to be both a technical and commercial success in this domain. C-by-C is characterized by an emphasis on a rigorous approach, the appropriate use of formal notations, and the static

verification of software, rather than relying on the inherently more risk-prone and expensive use of testing as the principal verification activity.

SPARK is a programming language that facilitates Correctness-by-Construction.

Its most important property is that it has a totally unambiguous semantics. This enables static analysis of the language to be both precise (i.e. the results really mean something and there's a very low "false alarm" rate) and efficient so the tool support is actually useful during development rather than just being used retrospectively.

SPARK is an annotated subset of Ada95, so the language inherits all of the support for commercial compilers, tools and training available from the wider Ada community. The "annotations" extend the subset's support for design-by-contract(tm), allowing packages and subprograms to specify their intended behaviour precisely at various levels of rigour and detail.

SPARK has over 10 years of industrial use, mainly in the European Aerospace and Rail industries, but remains less well known in the USA, although SPARK is actually recommended by NASA's own guidance for safety-critical software. This briefing will examine some of the most notable SPARK projects, including the Lockheed C130J, and how SPARK is effective in meeting standards such as DO-178B. Finally, we will consider some of the barriers (both actual and mythical) that may have hindered the adoption of SPARK in other projects. [...]

Dec 8-12 - ACM SIGAda 2002 Conference

From: jpwoodruff@irisinternet.net (John Woodruff)

Date: 18 Dec 2002 10:41:57 -0800

Subject: Re: SIGAda summary

Newsgroups: comp.lang.ada

[See also same topic in AUJ 23-4 (Dec 2002), pp.189-190. -- dc]

> Is there anyone who was at the SIGAda conference who would care to post a summary of anything interesting that was discussed?

Here are some of the items that interested me:

SIGAda Chair Currie Colket reports that there are 560 members in the SIGAda interest group. Attendance at the conference was just under 100.

Proceedings of the conference will be issued to SIGAda members in CD form.

Keynote speaker Robert Dewar described the license that Ada Core Technology (ACT) uses to release public versions of their compiler product. The license allows applications to use their runtime but never obligates a developer to release any product. ACT (the company) is steadily growing and makes a growing profit.

Bob Carey of Lawrence Livermore National Lab described the on going work on the controls software to operate the National Ignition Facility laser. NIF is a very large high-power laser that will focus 1.8 MegaJoules of energy on a target in a 25 nanosecond pulse to study nuclear fusion. The controls, written in Ada and Java, now comprise some 600 KSLOCs; initial integration testing is presently in progress. (I am a co-author of the report and a long-term coworker on NIF).

Rick Conn described the C-130J aircraft software. Lockheed has invested \$1.2 billion in the software product: about 5 MSLOC's of Ada in about 50 CSCI's. Their goal is to enhance customizability of the aircraft for a variety of customers.

Charles McKay, dean of the school of Science & Engineering of the University of Houston spoke in favor of yet-more-rigorous definitions of Ada and the systems engineering process in advance of mission and safety critical applications. He favors resuming work on a programming support environment (PSE) to capture all the development processes for safety-critical systems.

Martin Carlisle (USAF Academy) told about his revisions to GNAT to incorporate the Microsoft .NET paradigm (ability to distribute portable multi-language binary modules).

Tucker Taft outlined early plans for the next Ada language standard revision (Ada-0Y). The schedule calls for proposals to be complete early in 2003, the proposed text of the revision to be completed in spring 2005 and voting to be complete at the end of 2005.

Topics for probable revision: interface inheritance (ala Java); solve cyclic dependencies among types; pragma to declare intention to override (or avoid overriding) primitive operation in child package; standardize pragma Assert (leading to design-by-contact); possible syntax for prefix notation on object instances, like C++ Object.Op (params) notation; several issues for safety and security enhancements.

From: rod.chapman@praxis-cs.co.uk (Rod Chapman)

Date: 19 Dec 2002 03:56:49 -0800

Subject: A personal view of SIGAda

Newsgroups: comp.lang.ada

SIGAda 2002 - A brief report

Here are my thoughts on SIGAda 2002, held in Houston last week. Please note these are purely my personal thoughts and impressions. They only reflect a subset of the papers that caught my particular attention. There were a couple of sessions and many tutorials that I missed altogether. I leave it to others to fill in the blanks.

Overall: Generally, a good SIGAda. Much credit to all the SIGAda organising committee and program committee for such a well-run event. Turn-out was OK - about the same as the previous years at JHU and Minneapolis I think. In addition to all the "usual suspects", there were a few notable new faces such as NASA, TRW, Northrop Grumman, and Lawrence Livermore Labs.

Tutorials: I attended Michael McEvelley's tutorial on the Common Criteria. Good stuff - the CC is an important standard that the Ada community needs to be aware of. The ComSec world also appears to be a market where Ada could make a useful contribution. [ComSec = Communications Security. -- dc] I also gave a full-day SPARK tutorial. Very tiring!

Keynotes: A particularly strong set of key-note speakers this year.

The keynote from Robert Carey of Livermore Labs was amazing - they are building a thing called the "National Ignition Facility" (NIF), which is basically one of the world's biggest lasers. This is one of the last great "big physics" experiments left in the world. It will be used to generate small nuclear fusion experiments and so on. Check out www.llnl.gov/nif. Some metrics: peak power output 500 Terawatts; pulse energy 1.8 Megajoules; shot time 3 - 20 ns; laser amplification gain 3 million billion (!!!)

The control system is a large distributed network of machines - mostly off-the-shelf PCs, networking kit and embedded processors with some custom-built hardware. They expect it will be about a million lines of Ada and Java when finished. Most of the GUIs are Java, with the main control system being Ada. Oh, and when the shot fires, the whole thing is supposed to be synchronised to <20 _pico_seconds! Now that's what I call hard real-time!

Robert Dewar spoke about Open Source, Free Software, development processes (i.e. cathedral vs bazaar) and its effect on software reliability and security. Robert made the important point that "Free" or "Open Source" licensing and "Open" development processes have absolutely nothing to do with one another! Lively and topical as usual.

The keynote from Charles McKay was very good. Basically, he spoke passionately and strongly in favour of remembering Ada's "roots" (i.e. mission

and safety-critical systems) in the run up to the Ada0Y revision process.

Michael McEvelley did a good talk about the Common Criteria, security stuff, and why the Ada community has something to offer in the ComSec world (i.e. we know how to build really reliable software!)

Papers: Some good stuff. Some stuff I missed. Usual stuff from the vendors.

Two highlights for me were:

Richard Conn talked about the development processes used on the Lockheed Martin C130J software systems. Their development process is now very mature (they have a CMM Level 4) combining all sorts of best practices. The metrics presented by Richard suggest that the software development process is now one of the least error-prone activities in the aircraft's development.

VDot Santhanam from Boeing Wichita presented their "ZBra" Ada subset and compiler. This is an Ada subset, compiler and virtual-machine that is designed to be qualifiable as a development tool to the standards required by DO-178B Level A. This is a big breakthrough - a level A qualified compiler has never been attempted before to the best of my knowledge, so this is a significant announcement.

Next Year: San Diego! Excellent - let's hope the weather is better - well... couldn't be much worse... :-)

That's pretty much it... like I said, I hope someone else can fill in for the bits I missed...

From: JeLlyFish.software@gmx.net

(Vinzent Hoefler)

Date: 20 Dec 2002 13:45:42 -0800

Subject: Re: A personal view of SIGAda
Newsgroups: comp.lang.ada

Well, it was my first SIGAda and the reason I was there, were simply just private interests... but I'll try to comment a little bit.

[Removed many confirmations of points in posting above. -- dc]

> Michael McEvelley did a good talk about the Common Criteria, security stuff, and why the Ada community has something to offer in the ComSec world (i.e. we know how to build really reliable software!)

Yes, I have the feeling that especially more and more security people get interested into Ada (if they aren't already, of course). Not just from this keynote, also by my own experience.

> Papers: [...] Two highlights for me were:

Well, the paper about the Generic Genetic Algorithm was interesting, too. More in terms of teaching Software Engineering at University and experience of the students

with Ada than in the usual terms of the mission- and safety critical area.

All in all, I enjoyed the SIGAda2002 conference. Lots of interesting people. [...]

From: Ted Dennison

<dennison@telepath.com>

Date: Sat, 21 Dec 2002 00:33:29 GMT

Subject: Re: A personal view of SIGAda

Newsgroups: comp.lang.ada

[On security people getting/being interested in Ada: -- dc]

I've done ComSec Ada work before. I know the LMD/KP project was entirely Ada. I believe that's a piece of equipment that nearly every ComSec user is aware of. [LMD/KP = Local Management Device/Key Processor. -- dc]

Jan 9 - ACM DC SIGAda Meeting on Evolving Ada Bindings and Ada APIs

From: Currie Colket <colket@mitre.org>

Date: Mon, 6 Jan 2003 10:43:00 -0500

Organization: The MITRE Corporation

Subject: ACM DC SIGAda Chapter Meeting this Thursday in Washington DC

To: team-ada@acm.org

For those coming to the DC area this Thursday: our Next Meeting is scheduled for Thursday, 9 January 2003 at 7:30 P.M.

Mr. Clyde Roby, Institute for Defense Analyses, will be speaking on "Evolving Ada Bindings and Ada APIs" at the DC SIGAda Venue in McLean, Virginia. [...]

Abstract:

A workshop was held at SIGAda 2002 to address the management of Ada bindings and Application Program Interfaces (APIs). The purpose of that workshop was to identify a mechanism and begin to define a plan for the management of APIs for Ada. ISO/WG9 has requested that SIGAda and Ada-Europe propose such a mechanism for managing Ada bindings that are not covered via formal standards. Then ISO/WG9 can recommend these bindings as *defacto* standards, thus providing a process to register, and thus manage, bindings available to the Ada community without the formalism required by ISO. This workshop was well received at SIGAda 2002. There were many excellent suggestions made to improve a draft plan to manage Ada APIs.

Registered APIs are either Public or Private. Public APIs are those where a dedicated team assumes the responsibility for their maintenance and evolution. Changes are controlled via a consensus-based process.

Private APIs are those developed and maintained by a third party. With permission from the developer, these APIs are kept on the API web page as a convenience to the user community. In

both cases, useful artefacts will be provided on the API web page, when available. Besides compilable APIs, there will also be other valuable artifacts such as examples, tutorials, known problems, lessons learned, etc.

A major topic of discussion at the workshop was the management of both registered public and registered private APIs. There are many issues associated with each. It is proposed that a SIGAda API Working Group be responsible for this activity.

The workshop was extremely well received. The presentation on Thursday night will provide a summary of the workshop. Plans are already underway to implement this management plan. We are initially planning to use the Common Gateway Interface (CGI) as a test case for the procedures to manage Ada APIs. There will be other bindings of potential near term interest as well. Besides the CGI Bindings, other interfaces to which we have had recent presentations include: Windows Graphical User Interfaces and Hierarchical Data Format (HDF). This presentation could be valuable to all users of Ada bindings. Those interested in getting involved are especially welcome.

[...] Please visit the DC SIGAda Web site at <http://www.acm.org/sigada/locals/dc/> for additional information. [...]

Jeff Castellow, Chair, DC SIGAda

Jan 21 - U.S. Premiere of New Documentary Film "To Dream Tomorrow"

From: eugene@cse.ucsc.edu (Eugene Miya)

Date: 13 Jan 2003 22:50:49 -0800

Organization: Computer History Museum

Subject: Computer History Museum:

UPCOMING EVENTS

Newsgroups: [...],comp.lang.ada,[...]

Tuesday, January 21, 2003

U.S. Premiere of the new documentary film "To Dream Tomorrow" directed by Jo Francis and John Fuegi. The story of Ada Byron Lovelace and her work with Charles Babbage on the quest to build the first computer.

[...] Stanford University. FREE to the public. NO reservations required.

Abstract: In 1843, Ada Byron Lovelace moved beyond her illustrious predecessors Leibnitz and Pascal, and even her brilliant contemporary, Charles Babbage, to describe universal computing much as we understand it today. World authority on early computing devices, Doron Swade of the London Science Museum states in "To Dream Tomorrow": "Ada saw something that Babbage in some sense failed to see. In Babbage's world his engines were bound by number. What Ada Lovelace saw was that

numbers could represent entities other than quantity."

With the aid of Ada's descendant, the Earl of Lytton, the directors have done a fresh and comprehensive reading of Lovelace and Babbage material at Oxford's Bodleian Library, the Woking History Center, the British Library and in private collections. The film was shot at some of the most dazzlingly beautiful locations in England.

[...] Event URL: For directions and more information go to

http://www.computerhistory.org/events/lectures/ada_01212003/

[...]

From: tmoran@acm.org

Date: Wed, 22 Jan 2003 06:49:39 GMT

Subject: Ada movie, was Computer History

Museum, UPCOMING EVENTS

Newsgroups: comp.lang.ada

It was a rather good movie about Ada Augusta Lovelace. There were around 100 people pretty much filling the auditorium. One of the first questions was about the Ada language, and the response from the movie producer, unfortunately, was that he was told at U of Maryland that they don't currently teach Ada. So a little later I pointed out that since Babbage's original impetus for the Difference Engine was the bugginess of math tables, it was fitting that applications requiring high reliability are actively using the Ada language.

Feb 6 - Ada-Belgium Event on Testing Tools for Embedded Applications

From: dirk@cs.kuleuven.ac.be (Dirk Craeynest)

Date: 6 Feb 2003 22:33:30 +0100

Organization: Ada-Belgium, c/o Dept. of Computer Science, K.U.Leuven

Subject: Ada-Belgium - PolySpace presentation - Tue 2003-02-25 20:00

Newsgroups:

comp.lang.ada,fr.comp.lang.ada,be.com.programming,nl.comp.programmeren

Ada-Belgium is pleased to announce a technical presentation by Marc Lalo of PolySpace Technologies, France.

"Next generation testing tools for embedded applications: Apply abstract interpretation to automatically detect run-time errors at compilation time"

Tuesday, February 25, 2003, 20:00, at the U.L.B., Department of Computer Science, [...], B-1050 Brussels (after the Ada-Belgium 2003 General Assembly at 19:00).

<http://www.cs.kuleuven.ac.be/~dirk/ada-belgium/events/local.html>

[...]

Abstract

Real time software applications may be incorrect due to three main reasons: run-time errors (latent faults in the code); non respect of the temporal requirements; non respect of the functional requirements.

Our talk will focus on the first solution to automatically detect run-time errors at compilation time (arithmetic overflows, arithmetic exceptions (division by zero, square root of a negative number), out-of-bounds array access, access conflicts to shared data, read access to a non-initialized variable, illegal dereferenced pointer).

In real time software applications, the following strategies are used to deal with run-time errors: fault-tolerant systems; fault detection, based on intensive testing and code review. Unfortunately, due to the increase of software size and complexity, the cost of testing and code review dramatically increases. PolySpace Technologies has developed the first solution based on abstract interpretation techniques for an earliest and automatic detection of run-time errors in Ada and C applications.

We will report results obtained in the embedded systems industry and more specifically by our customers involved in the development of safety and mission critical Ada software: aerospace, railway transport and defense.

More information: PolySpace Technologies - <http://www.polyspace.com/>

[...]

Jun 16-20 - Ada-Europe 2003 Conference

From: Jean-Pierre Rosen
<rosen@adalog.fr>

Date: Wed, 5 Feb 2003 14:52:11 +0100

Subject: Ada-Europe program
Newsgroups: comp.lang.ada

The preliminary program, as well as the tutorials program, is now available from the conference page at: <http://www.ada-europe.org/conference2003.html>. We hope that it will convince you that attending the conference is a MUST!

J-P. Rosen, rosen@adalog.fr,
<http://www.adalog.fr>

Dec 7-11 - ACM SIGAda 2003 Conference

From: Clyde Roby <roby@ida.org>

Date: Mon, 3 Feb 2003 13:03:58 -0500

Subject: SIGAda and Conferences reminder
To: SIGAda-announce@acm.org

[...] Don't forget about our upcoming Ada-related conferences:

SIGAda 2003, San Diego, California USA, 7-11 December 2003.

<http://www.acm.org/sigada/conf/sigada2003/>

[...]

Clyde Roby, SIGAda Secretary

Ada and Education Course and Training Material in French

[Extracts translated from French.

Previously included pointers are not repeated here; see also the same topic in AUJ 23-4 (Dec 2002), p.191, and AUJ 23-3 (Sep 2002), p.128. -- dc]

From: Bruno Kuper <bkuper@club-internet.fr>

Date: Sat, 23 Nov 2002 08:20:44 +0100

Subject: Re: Qqn connait un site qui donne des tutoriaux pour Ada?

Newsgroups: fr.comp.lang.ada

There are several sites where one finds Ada tutorials. [...] If you type Ada in a search engine you will find tens in French of them. Especially in Switzerland, [...] Thanks to our Swiss friends to do so much for our language.

From: GuillaumeM <autre>

Date: Sat, 7 Dec 2002 20:03:21 +0100

Subject: Re: je commence a apprendre Ada
Newsgroups: fr.comp.lang.ada

Here are some links.

<http://boole.imag.fr/PLACARD/ECOLE1/lp1-ricm/>

["Langages et Programmation" page at the Institut d'Informatique et Mathematiques Appliquees de Grenoble, with among others notes for the course "Programmation en langage Ada". -- dc]

<http://lithwww.epfl.ch/teaching/cmp/polycop.html>

[Course text for the "Cours de programmation Ada" at the Computer Science Theory Laboratory (LITH) of the Ecole Polytechnique Federale de Lausanne (EPFL). -- dc]

From: Jean-Pierre Rosen

<rosen@adalog.fr>

Date: Fri, 27 Dec 2002 10:07:27 +0100

Subject: Re: Debutant
Newsgroups: fr.comp.lang.ada

<http://www.adalog.fr/adaweb.htm>

[Page in French with an extensive "catalogue of the most important Ada resources available on Internet (except compilers)". -- dc]

From: Thomas De Contes

<d.l.deContes@iFrance.com>

Date: Mon, 13 Jan 2003 17:57:45 +0100

Subject: Re: [ada-france] Manuels d'auto-formation en Ada

To: liste ada-france <ada-france@ada-france.org>

<http://baptiste.malguy.net/contrib/ada/>

["Ada 95 Tutorial" in French, as a compressed HTML archive. -- dc]

From: Daniel Feneuille <feneuille@univ-aix.fr>

Date: Tue, 14 Jan 2003 11:35:51 +0100

Organization: IUT Aix

Subject: Re: [ada-france] Manuels d'auto-formation en Ada

cc: ada-france@ada-france.org

To learn Ada (except real time) you can consult and download the courses I teach at the IUT at Aix-En-Provence:

http://libre.act-europe.fr/french_courses

[...] With regard to French books the most complete are:

1) "Programmation Séquentielle avec Ada 95" of P. Breguet and L. Zaffalon at the Presses Polytechniques et Universitaires Romandes which does not treat "real time", but it is supplemented by

2) "Programmation Concurrente et Temps Réels avec Ada 95" same authors and editor,

3) "Programmer en Ada 95" de John Barnes at Addison Wesley.

With regard to CDROMs, I distribute one to my students containing texts and free software, but you can check out the one proposed by Stéphane Rivière: "Aide", stephane@rochebrune.org. [See also "Ada Starter CD-ROM for MS-Windows" further in this news section. -- dc]

Ada as a First Language - an Anecdote

From: Sébastien Martin

<Sebastien.Martin2@free.fr>

Date: Sat, 23 Nov 2002 20:47:45 +0100

Subject: Re: nb aléatoire

Newsgroups: fr.comp.lang.ada

[Extracts translated from French. -- dc]

I will simply tell you an anecdote: in my school (Fiifo), the language used to learn programming in the first year is Ada. Most of the students complain about this programming language which "annoys" them constantly with its constraints. Then in second year, we learn C (as well as COBOL, but good...) and its joys. Curiously, at the time of our algorithmic project, almost all students chose to implement it in Ada...

Sébastien Martin, Student Fiifo 2 [Fiifo = Formation d'Ingénieurs en Informatique de la Faculté d'Orsay, France. -- dc]

From: Sébastien Martin

<Sebastien.Martin2@free.fr>

Date: Sun, 24 Nov 2002 13:02:53 +0100

Subject: Re: nb aléatoire

Newsgroups: fr.comp.lang.ada

My example did not want to show a "superiority" of Ada over C [...], but to illustrate that the more significant "constraints" of Ada actually made the language simpler and more effective to use, since it is a language designed from the start to be used by a human being, and not to be understood efficiently by a

machine - and that the students finally appreciated it for that. [...]

[And from a later message: -- dc]

[...] I didn't say that [C] was unusable, I simply noticed many students had preferred Ada over C for the implementation of their algorithm, because Ada appeared easier to use.

More on Ada as a First Language

From: john@assen.demon.co.uk (John McCabe)

Date: Thu, 02 Jan 2003 10:04:13 GMT

Subject: Re: Why is Ada a good choice for an ambitious beginner to programming
Newsgroups: comp.lang.ada

> The advantage of starting in Ada is that you won't acquire bad habits. This is really important. The disadvantage is that Ada is difficult. Some people may tell you otherwise, they are geniuses creating software for devices that fly to the cosmos for a few years now.

Bill Findlay responded:

> 400 not-especially-endowed-with-genius CS1 students have learned to program using Ada 95 at Glasgow University every year since 1996.

Coming from the bloke who co-wrote one of the definitive Pascal manuals of the late '70s, early '80s, I would suggest this is quite an endorsement for Ada over Pascal.

For what it's worth, I was taught Pascal at Glasgow University in around 1983 (as part of an Electronics Engineering degree).

From: Bill Findlay

<BillFindlay@blueyonder.co.uk>

Date: Thu, 02 Jan 2003 21:32:26 GMT

Subject: Re: Why is Ada a good choice for an ambitious beginner to programming
Newsgroups: comp.lang.ada

Ada is easier for beginners than Pascal, because its syntax and semantics are much more consistent. In fact, it's just a better language, full stop. The Hoare/Dijkstra barb about Pascal being an improvement on its successors is nonsense. Hoare's list (in "The Emperor's Old Clothes") of the supposed advantages of Pascal can be seen, in retrospect, as a catalogue of the reasons for Pascal's demise.

> For what it's worth, I was taught Pascal at Glasgow University in around 1983 (as part of an Electronics Engineering degree).

It took ten years for EE to switch from Fortran to Pascal. Fifteen years further on, when CS adopted Ada 95, they tried to take EE with them, but without success. AFAIK they are still stuck with Pascal. (In fairness to EE, they did see through C,

and did not want it taught to their beginners.)

It was interesting to stand in the lab and watch CS (Ada) and EE (Pascal) beginners working side by side. One day, about week 6 of term, the CS students (using GNAT and AdaGraph) had nearly all successfully written a program that animated balls bouncing around in a box. The EE students (using Turbo) were nearly all grappling unsuccessfully with a trivial text-oriented read-process-output loop.

From: Bill Findlay

<BillFindlay@blueyonder.co.uk>

Date: Sun, 05 Jan 2003 00:50:34 GMT

Subject: Re: Why is Ada a good choice for an ambitious beginner to programming
Newsgroups: comp.lang.ada

> You've convinced me; but the CS students (using GNAT and AdaGraph) were enrolled in a formal course ...

So were the EE students. In fact, they were enrolled in a programming module taught by CS. (I was not teaching either group BTW).

But I don't want to make too much of this. Certainly there were other significant differences between the groups that accounted for much of the disparity, but I'm quite sure the EE group would have made better progress with the easier language (i.e. Ada). That's the irony. [...]

> [...] exercises that inject some colour and fun into the process of learning (e.g., the ball/box graphics exercises).

If you are using GNAT on a Wintel system, all you need to get going is the AdaGIDE IDE and the AdaGraph library package (and some neat ideas 8-).

From: john@assen.demon.co.uk (John McCabe)

Date: Mon, 06 Jan 2003 10:05:30 GMT

Subject: Re: Why is Ada a good choice for an ambitious beginner to programming
Newsgroups: comp.lang.ada

[On the remark that "Ada is easier for beginners than Pascal, because its syntax and semantics are much more consistent." -- dc]

I agree with this, but you definitely need to go for one of the books written especially for teaching Ada *and* programming together. Norman Cohen's book, "Ada as a second language", for example is not designed to do that, yet is (IMO) one of the best Ada books around. Ada comes across as being such a large language that, without a good reference guide to what you *need* to know, it could be very easy for a beginning programmer to get lost in all the stuff you don't need to know. [...]

Feedback on Ada Distilled Online Book

From: Richard Riehle

<richard@adaworks.com>

Date: Thu, 09 Jan 2003 18:58:38 -0800

Organization: AdaWorks Software Engineering

Subject: Re: Why is Ada a good choice for an ambitious beginner to programming
Newsgroups: comp.lang.ada

One reason I wrote Ada Distilled (downloadable from adaic.org and adapower.com) was to focus on what the newbie needed to know to get started with Ada without plowing through a massive amount of prose and long programs. I believe everyone should have a copy of Norm Cohen's book if they are serious about Ada, just as they should have a copy of John Barnes' book to fill in a lot of gaps as well as to see some uniquely conceived examples.

The feedback I have received on Ada Distilled is that it has made Ada more immediately accessible to at least some experienced programmers than those more complete and arguably much better books by Cohen and Barnes. Everyone learns in a different way. We can always use more pedagogical material. I have been considering doing a book that focuses only on Ada for numerics applications, one that demonstrates the usage of the many kinds of numeric attributes in actual programming examples. Alas, the press of other commitments has delayed the creation of that work.

Ada Accessible for New Students

From: Richard Riehle

<richard@adaworks.com>

Date: Tue, 07 Jan 2003 05:51:33 -0800

Organization: AdaWorks Software Engineering

Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

[...] I just returned from presenting a two week software engineering seminar in Asia. I used Ada to demonstrate my programming examples. The students, none of whom had seen Ada before, found the examples easy to read, understand, and ask questions about. If I had been using C++ for a similar audience, one that had never seen C++ before, I suspect I would have had a classroom full of confused students. Even my example using an Ada generic component was easy to follow for these students, even though English was not their native language.

Some of them plan to download GNAT and experiment with Ada. Everyone of them agreed that C++ is so confusing that they spend a huge amount of time correcting errors that they wish had been

caught by the compiler. Many had already rejected C++ in favor of Java, Python, and other languages because they found C++ so difficult.

Ada can be made accessible to new students. We simply need to learn how to present what the students need when they need it, and to show them simple solutions they can use to work out their own problems.

From: Richard Riehle
<richard@adaworks.com>
Date: Thu, 09 Jan 2003 18:38:55 -0800
Organization: AdaWorks Software Engineering
Subject: Re: Hijacking a Thread
Newsgroups: comp.lang.ada

[...] I was in Wuhan, China for the past two weeks presenting some seminars at Wuhan University. For most of the students, it was the first time they ever saw Ada. Amazing how easy it is for people who have never seen Ada before to read and understand the key points.

ACT alert. Do expect some downloads of GNAT from Hubei Province in China.

Ada-related Resources

CORBA Resources

From: Victor Giddings
<victor.giddings@ois.com>
Date: Tue, 05 Nov 2002 19:12:43 GMT
Organization: Objective Interface System
Subject: Re: Distributed programming in heterogenous platforms
Newsgroups: comp.lang.ada

> Before bumping my head into the project, I like to gather as much information about distributed programming in heterogenous platforms.

CORBA is another technology that explicitly deals with platform heterogeneity, as well as language heterogeneity. We have a number of getting started links on our web site:

<http://www.ois.com/resources/corb-1.asp>

[With among others in the section "Objective Interface's Presentations & Papers": an explanation how Ada should be considered for real-time applications in "Beyond C++ and Java: Extending the Reach of Ada through CORBA" in the May 2001 version of the COTS Journal Online, and the presentation "ORBexpress for Ada 95 - CORBA: How IT Works". -- dc]

Victor Giddings, Senior Product Engineer, Objective Interface Systems

Useful Resources for Ada 83 to 95 Porting

From: Robert A Duff
<bobduff@shell01.TheWorld.com>

Date: Thu, 7 Nov 2002 15:44:56 GMT
Subject: Re: differences between Ada83 and Ada95
Newsgroups: comp.lang.ada

> I'm looking for a complete list of differences between Ada83 and Ada95 (not only upward compatibility).

You want the Annotated Ada Reference Manual (AARM). It's available at:

<http://www.ada-auth.org>.

Each section contains headings: inconsistencies with Ada 83 (cases where a legal Ada 83 program has different run-time semantics in Ada 95); incompatibilities with Ada 83 (cases where a legal Ada 83 program is illegal in Ada 95); extensions to Ada 83 (cases where a legal Ada 95 program was not legal in Ada 83).

When we designed Ada 95, we were pretty careful to document every change, and categorize them under the above headings. We missed a few obscure ones by accident, but they are almost 100% complete.

From: Lauper_Marc_Eig@Hotmail.com
(Lauper Marc)
Date: 29 Nov 2002 14:26:00 -0800
Subject: Re: Pragma BUILT_IN
Newsgroups: fr.comp.lang.ada

[In response to a question about implementation-dependent pragmas in old Ada 83 code: -- dc]

The following document seems to refer to this, it might perhaps help you:

<http://archive.adaic.com/pol-hist/history/9x-history/reports/pragma-Nov89.txt>

[This is an "Ada 9X Project Report" entitled "A Study of Implementation-Dependent Pragmas and Attributes in Ada" from November 1989. -- dc]

Interesting Historical Document

From: John English <je@brighton.ac.uk>
Date: Wed, 18 Dec 2002 12:39:03 +0000
Organization: University of Brighton
Subject: Re: Interesting and historical document
Newsgroups: comp.lang.ada

[A scanned version of Edsger Dijkstra's (in)famous critique of the Green language (which evolved into Ada) is available at

<http://www.cs.utexas.edu/users/EWD/ewd06xx/EWD660.PDF>

-- dc]

Interesting indeed. Having never seen the original Green proposal, it seems that quite a lot got changed before it reemerged as Ada. Does anyone have a copy of the original Green proposal they could put online as well? (Also Red and the other colour -- was it Yellow?)

John English, Senior Lecturer, Dept. of Computing, University of Brighton,

<http://www.it.bton.ac.uk/staff/je>

From: Robert A Duff
<bobduff@shell01.TheWorld.com>
Date: Wed, 18 Dec 2002 15:20:58 GMT
Subject: Re: Interesting and historical document
Newsgroups: comp.lang.ada

There were four: Red, Green, Blue, and Yellow. At some point, they narrowed it down to Red and Green. Then just Green, which evolved into Ada.

From: John English <je@brighton.ac.uk>
Date: Thu, 19 Dec 2002 10:12:32 +0000
Organization: University of Brighton
Subject: Re: Interesting and historical document
Newsgroups: comp.lang.ada

The preliminary spec was published in SIGPLAN Notices in 1979, but it was already called Ada by then. However, it was reasonably similar to the final form of the standard by then. Dijkstra's comments about Green make me think that Green must have been a very different beast indeed from what was eventually published in 1979, and I'd be curious to know more about it. It would also be nice to see the other proposals -- IIRC, Red was based on Algol 68, but that's all I know about it.

Ada 200x Resources

From: Randy Brukardt
<randy@rrsoftware.com>
Date: Fri, 27 Dec 2002 13:32:38 -0600
Subject: Re: Ada 200x
Newsgroups: comp.lang.ada

> Is there a website for Ada200x?

Not per se. What material there is is on the ARG site (www.ada-auth.org).

Randy Brukardt, ARG Editor

From: Robert A Duff
<bobduff@shell01.TheWorld.com>
Date: Fri, 27 Dec 2002 20:55:06 GMT
Subject: Re: Ada 200x
Newsgroups: comp.lang.ada

Look at the amendment AI's to see the various proposals. In particular, search for "!class amendment" and "!status Amendment".

Ada Forum in German

From: gdemont@hotmail.com (Gautier)
Date: 6 Jan 2003 12:52:44 -0800
Subject: Ada forum in German
Newsgroups: comp.lang.ada, comp.lang.misc

There is a new forum in German about the Ada programming language on:

<http://www.ada95.ch/forum/forum.html>

Ada Web Site in Russian

From: *Dmitriy Anisimkov*
<*anisimkov@yahoo.com*>

Date: *Mon, 27 Jan 2003 22:27:56 +0600*

Subject: *An: www.ada-ru.org*

Newsgroups: *comp.lang.ada*

<http://www.ada-ru.org>

Welcome to new site about Ada in Russian.

There is no content in English for now, because our main goal is to represent Ada for the Russian programmers. There are enough Ada sites in English in the world and <http://www.ada-ru.org> has links over there.

We invite Russian speaking people to intent our site. We appreciate any comments and new Russian stuff to populate this site.

Note, this site is powered by the Ada Web Server. <http://libre.act-europe.fr/aws>

New Ada-France Web Site

From: *Lionel Draghi*
<*lionel.draghi@free.fr*>

Date: *Mon, 03 Feb 2003 22:14:55 +0100*

Subject: *L'association Ada-France a un nouveau site WWW*

Newsgroups: *fr.comp.lang.ada*

[Translated from French: -- dc]

The team of Ada-France is happy to announce the availability of her new WWW site. The use of the SPIP publication software allows in particular:

- to quickly access the Ada topics in French (those who do not have time to follow the newsgroups *comp.lang.ada* and *fr.comp.lang.ada* will appreciate);

- to function in collaborative mode: the readers can also become writers and propose articles, which will be validated by the editors.

The search engine integrated into the site makes it possible to find information easily. The use of key words gives immediately access the articles with identical subjects. The services offering "stages" and employment will be reintroduced soon in a new form.

The site is accessible at the usual address right now: <http://www.ada-france.org/>.

Do not hesitate to send us your remarks at the address www@ada-france.org

For the Ada-France organization,
Lionel Draghi et Samuel Tardieu.

Ada-related Tools

Ada Components and APIs Working Group

From: *Randy Brukardt*
<*randy@rrsoftware.com*>

Date: *Mon, 6 Jan 2003 14:52:36 -0600*

Subject: *Re: Anybody in US using Ada?*

Newsgroups: *comp.lang.ada*

>[...] let's get a group to organize an "Ada Package Distribution".

SIGAda is supposedly working on setting up such a thing. Hopefully, they'll succeed, because they are the right sort of organization to do it: independent of the vendors; not a standards body (which would take years) yet they have visibility and clout to get people to pay attention.

[See also "Jan 9 - ACM DC SIGAda Meeting on Evolving Ada Bindings and Ada APIs" earlier in this news section, and the web site for the Ada Application Programming Interfaces Working Group (APIWG) at

<http://www.acm.org/sigada/locals/dc/>

-- dc]

From: *Randy Brukardt*
<*randy@rrsoftware.com*>

Date: *Tue, 7 Jan 2003 13:15:11 -0600*

Subject: *Re: Anybody in US using Ada?*

Newsgroups: *comp.lang.ada*

> SIGAda is the right forum for establishing all sorts of things as being "Conventional Ada" and you're 100% right about its advantages compared to the vendors or standards body. However, I think it could benefit if it had some acceptance & endorsement by the vendors and standards body so that everyone saluted the flag concerning the end results. [...]

I can't speak for the vendors, of course. As far as the standards body goes, that's where the idea originated (inside of WG9). SIGAda volunteered to try to implement it, because it is out of scope for WG9. The hope is that some of the packages pseudo-standardized by SIGAda could be picked up and added to the standard in the future.

I suspect that getting the vendors to do anything will be a hard sell. The idea as I recall it (I wasn't at either the recent SIGAda meeting or the most recent WG9 meeting, so the plan may have changed) was to set up a place where "certified packages" could be accessed. Such software would already have been tested on a variety of compilers, so it ought to work out of the box for you. Java supposedly has something similar.

The problem with most of the stuff on the net is that it has only been tested with whatever compiler the author has. And it often doesn't work anywhere else. Our experience with Claw is that you really have to compile and test it on each compiler if it does anything interesting at all, because some things won't work, and some compilers have bugs that allow illegal code to be compiled, and so on. If this stuff works well and is popular, the vendors will pick up on it.

Booch Components

From: *Simon Wright*
<*simon@pushface.org*>

Date: *Sun, 12 Jan 2003 12:26:06 GMT*

Subject: *Booch Components 20030111*

To: *team-ada@acm.org*

This release has been uploaded to

<http://www.pushface.org/components/bc/>

and is mirrored at

<http://www.adapower.net/booch/>.

Major features since 20020602:

Licence changes: The library is now released under the GNAT-modified GPL. My test and demonstration code is released under the GPL. Contributed code's licences remain unchanged.

Interface changes

Constraints: For Bags, Maps and Sets in each form, and in all Bounded forms, the basic container is now unconstrained with the previous container (Bag etc) implemented as a constrained subtype. The intention is that users shouldn't need to change existing source.

Iterators: It used to be possible to create an empty Container, create an Iterator over it, populate the Container, and then just use the Iterator. If you do that now, the Iterator will typically be "complete" (ie, `Is_Done` will return True) until you Reset it.

Concurrency: The slender support for concurrency (Guarded and Synchronized forms) has been removed.

[See also same topic in AUJ 23-3 (Sep 2002), pp.129-130. -- dc]

Mats Weber's Ada Component Library

From: *matsw@bluewin.ch (Mats Weber)*

Date: *17 Jan 2003 11:56:17 -0800*

Subject: *New location for my Ada component library*

Newsgroups: *comp.lang.ada*

The new web location for my Ada components is

http://mats.weber.org/ada/mw_components.html.

Is anybody still using these?

[See also "Mats Weber's Ada components" in AUJ 20-1 (Apr 1999), pp.14-15. -- dc]

My PHD thesis has also moved, there are links to it from my home page:

<http://mats.weber.org/>.

From: *Jeffrey Creem*
<*jeff@thecreems.com*>

Date: *Sat, 18 Jan 2003 14:52:22 GMT*

Subject: *Re: New location for my Ada component library*

Newsgroups: *comp.lang.ada*

I still occasionally use them. Mostly bags. There are a lot of standard Ada library

initiatives that pop in here from time to time. Maybe at some point I will switch but it is hard to beat the simplicity of most of your stuff for day to day quick and dirty programs.

GNAT 3.15p Binaries for OS/2

From: DWParsons@T-Online.de (Dave Parsons)
Date: Sun, 24 Nov 2002 17:52:40 +0100
Subject: ANN: GNAT 3.15p binary for OS/2 available
Newsgroups: comp.lang.ada

Following the release of GNAT 3.15p by ACT last week, I have now compiled a binary for OS/2 V4 / eCS.

This is again based upon EMX 9d with GCC 2.8.1. See Arnaud Charlet's post and the readme for more details. It is currently available as

<http://dwp Parsons.bei.t-online.de/gnat-3.15p-os2-bin-20021124.zip>

and it is expected that it will be mirrored by cs.nyu.edu in

<ftp://cs.nyu.edu/pub/gnat/3.15p/contrib/os2>

GNAT for Mac OS X

From: jim hopper
<hopperj@macconnect.com>
Date: Thu, 09 Jan 2003 23:14:50 -0500
Subject: Re: GCC and Ada in OS X 10.2.3?
Newsgroups: comp.lang.ada

> Does anyone know of an Ada Compiler package for Mac OS X 10.2.3?

[See <http://www.adapower.net/macOS/> and also "GNAT and GCC 3 for Mac OS" in AUJ 23-3 (Sep 2002), p.132. -- dc]

I am one of the folks working on it. Join in our mailing list and ask questions if you have them. The compiler is very robust, I know of two commercial packages which have been successfully ported to OS X using the compiler (one of them our radar simulation), and there is one other large commercial app being ported now using it.

Also we have integrated Ada into Apples IDE called project builder and their interface builder GUI development. Ada aware GDB, and much more.

From: jim hopper
<hopperj@macconnect.com>
Date: Tue, 28 Jan 2003 18:33:59 -0500
Subject: Re: GCC and Ada in OS X 10.2.3?
Newsgroups: comp.lang.ada

[On an installation question: -- dc]

You are only supposed to pick one version of the compiler not both ;-). The wavefront version is actually the one you want. If you did a successful install of the wavefront you are there. Well there are other things you can install like Ada aware gdb (on the site), integration to

Apples IDE and bindings to their GUI (also on the site).

[...] Best place to ask these questions is the gnat mac mailing list. You can sign up on our website <http://adapower.net/macOS/>. We have a very helpful group and questions are answered way more promptly than we can manage here.

Welcome to Ada on the Mac.

GNAT 3.14/15p Binaries for HPUX-11

From: Warren W. Gay VE3WWG
<ve3wwg@cogeco.ca>
Date: Sun, 19 Jan 2003 18:00:31 -0500
Subject: Announce: HPUX-11 Binaries for GNAT-3.14p/3.15p Available (and HOWTO)
Newsgroups: comp.lang.ada

Since there has been no HPUX binaries emerging in the public (that I know of) for GNAT-3.14p/3.15p, I took on the "gauntlet" myself, and after some struggling, succeeded. The process is amazingly problematic under HPUX, and without root access, becomes even more of a challenge.

Because both of these compilers are still based upon GCC 2.8.1, they and the code they generate cannot link against shared libraries under HPUX. We'll have to stay tuned for the new crop of GCC 3.x compilers for that. :-)

I have created a PDF document that describes in detail how to get past all of those build problems, without the need for root access. The document is available on my website.

[<http://home.cogeco.ca/~ve3wwg> -- dc]
 I also have binary distributions for GNAT-3.14p and 3.15p available to share, but I don't have enough space on my website to host them there (they are about 16MB each).

These install under /opt/gnu/gnat-3.14p (or 3.15p), but with the help of environment variables can be installed anywhere else (including your home directory). Email me if you want them, and have a ftp site for me to upload them to. They are too large to email.

I would also like to contribute these to the [ftp.cs.nyu.edu](ftp://cs.nyu.edu) ftp site, but I did not see any documentation on who I should contact. If someone can point me to the correct party, I'll see if I can make them available there.

For the PDF document, see my website (Click on the Software 'n Tools picture).

From: Warren W. Gay VE3WWG
<ve3wwg@cogeco.ca>
Date: Tue, 21 Jan 2003 14:53:57 -0500
Subject: Re: Announce: HPUX-11 Binaries for GNAT-3.14p/3.15p Available (and HOWTO)
Newsgroups: comp.lang.ada, comp.sys.hp.hpux

No longer a problem. Graham (graham@tecknet.net) is kindly hosting [the binary distributions] now at the following location:

<http://www.vmsdatanet.co.uk/gnat>

Later on, I'll post links from my own web site to these. [Done. -- dc]

Ada Compiler Vendors

From: Tucker Taft <stt@sofcheck.com>
Date: Mon, 13 Jan 2003 14:14:29 -0500
Organization: SofCheck, Inc.
Subject: Re: [Ada-Comment] Ada Compilers

To: ada-comment@acm.org

[Someone in DoD asked: -- dc]

> Since the Government is no longer mandating the use of Ada and programs have mostly migrated to other platforms, what happened to all the Ada software that the government purchased? The reason I am asking this question, we have a number of legacy systems and sometimes they ask that we purchase Ada products, such as an Ada compiler. Do inventories exist for Ada products? Thanks--

I presume you will get several answers. In fact there are still a significant number of projects using Ada (e.g JTF, International Space Station, NYC subway, Channel tunnel, many nuclear power systems, etc.). There are several companies still actively selling and supporting Ada compilers (for a variety of hosts and targets), including:

AdaCore Technologies: <http://www.gnat.com>

Aonix: <http://www.aonix.com>

DDCI: <http://www.ddci.com>

Green Hills Systems: <http://www.ghs.com>

Irvine Compiler Corp: <http://www.irvine.com>

Rational (now part of IBM):

<http://www.rational.com>

RR Software: <http://www.rrsoftware.com>

SofCheck, Inc. (formerly Intermetrics/AverStar/AverCom):
<http://www.sofcheck.com>.

There are a number of websites devoted to Ada, including the Ada Information Clearinghouse, sponsored by the Ada Resource Association, which may be your best starting place: <http://www.adaic.org>

I hope this answers your questions.

Tucker Taft, President, SofCheck, Inc.,
 President, Ada Resource Association

From: dewar@gnat.com (Robert Dewar)
Date: Tue, 14 Jan 2003 05:26:25 -0500 (EST)

Subject: Re: [Ada-Comment] Ada Compilers

To: ada-comment@acm.org, ada-comment@ada-auth.org

> "programs have mostly migrated to other platforms"

That's certainly not what we see. yes, there is some migration, but then there are completely new projects as well.

New OpenGL Binding

*From: David Holm
<david@realityrift.com>
Date: Sat, 28 Dec 2002 06:32:41 GMT
Subject: New Ada binding to OpenGL
Newsgroups: comp.lang.ada*

As the original binding to OpenGL seems to be pretty dead (the only places to find it is in AdaSDL and GtkAda) I started a new project. My bindings are not based on the old ones. The current version is based on MesaLib 5.0. At the moment there are only thin bindings but the idea is to implement a thick binding as soon as the thin one works as it should.

The binding is located here:
<http://adaopengl.sourceforge.net/>

[...] I released a preview version (0.1). Please look through it and send constructive feedback to me. Also check the Readme before mailing me as it contains some implementation notes I scribbled down.

[See also "OpenGL Bindings and Demos" in AUJ 22-2 (June 2001), pp.72-73. -- dc]

*From: David Holm
<david@realityrift.com>
Date: Sat, 11 Jan 2003 10:46:51 GMT
Subject: AdaOpenGL 0.9 released
Newsgroups:
comp.lang.ada,comp.graphics.api.opengl*

I just finished uploading AdaOpenGL 0.9 to sourceforge. The URL is <http://adaopengl.sourceforge.net/>. This is the first stable release of AdaOpenGL, feedback is welcome as usual.

Changelog:

New features: glut support; replaced all uses of "access GL*" in function/procedure declarations with GL*Ptr types; added glut example; added SDL example.

Bugfixes: you can now pass Ada null's to gl* functions (that support it); removed unnecessary overloaded functions that used System.Address since you can use 'Unchecked_Access to pass an aliased variable as a pointer; stronger typing; glX fixed, requires a patched AdaBindX 0.7.2, or a newer version (not available at the moment); glX example now working; a lot of sourcecode cleanups.

DirectX Examples

*From: David Botton <David@Botton.com>
Date: Wed, 22 Jan 2003 00:40:47 -0500
Subject: DirectX 7 with Ada
Newsgroups: comp.lang.ada*

Do to popular demand, I have created an example of using GWindows/GNATCOM with DirectX 7

at:
<http://www.adapower.com/gwindows/ddraw1.zip>

This is just a simple example, but it is a starting point.

[See also "DirectX" in AUJ 22-3 (Sep 2001), p.141. -- dc]

*From: David Botton <David@Botton.com>
Date: Fri, 24 Jan 2003 12:49:42 -0500
Subject: DirectX 8 with Ada
Newsgroups: comp.lang.ada*

I have created a simple DirectX 8 example using GNATCOM/GWindows at: <http://www.adapower.com/gwindows/dx8-prj1.zip>

Create the DirectX8 binding using the `_latest_` version of `gbmanager.exe` (the GUI ActiveX/COM/DCOM binding manager, part of GWindows) at

<http://www.adapower.com/gwindows/gbmanager.exe>

(I will be updating GNATCOM soon with the fixes, etc that are needed for DX8 and some other newer typelibs).

I'd love to hear if any one does anything with this.

Tty Based GUI Packages and Examples

*From: John Stoneham
<captjameskirk@yahoo.com>
Date: Sat, 9 Nov 2002 22:20:56 -0600
Subject: Re: Using NCurses with Ada
Newsgroups: comp.lang.ada*

The GNU NCurses 5.2 package has built-in bindings for Ada95. I've never used it, but I can tell you there is a "sample-curses_demo.adb" (along with several others) in the source distribution.

*From: Dale Stanbrough
<dstanbro@bigpond.net.au>
Date: Sun, 10 Nov 2002 21:59:02 GMT
Subject: Re: Using NCurses with Ada
Newsgroups: comp.lang.ada*

I've written a simple forms based package that can be found at

<http://goanna.cs.rmit.edu.au/~dale/software/index.html>

which uses curses (not ncurses).

*From: Fulcanelli
<fulcanelli@tuxfamily.org>
Date: Mon, 11 Nov 2002 16:31:07 +0100
Subject: Re: Rafrachir fenetre sous DOS
Newsgroups: fr.comp.lang.ada*

[Translated from French: -- dc]

You can go to this page, it is a package (termi.ads and termi.adb) which manages a terminal window, I tested it under linux OK, under Windows 95 and 98 there are some display problems, not tested yet under XP pro.

<http://perso.club-internet.fr/ariffart/ada/ada05.html#ada055>

VAD - Visual Ada Developer Ported to ObjectAda

*From: Stas <dulman@attglobal.net>
Date: Mon, 6 Jan 2003 07:43:11 +0200
Subject: VAD (Visual Ada Developer) port to Aonix ObjectAda.
Newsgroups: comp.lang.ada*

VAD (Visual Ada Developer) is now ported to Aonix ObjectAda 7.2 compiler. You may build project from VAD generated Ada sources. Unfortunately I have only Windows compiler and can not build port for Unix (Linux).

You may download libraries and sample from

<http://www.websamba.com/ldulman/vad.htm>

or directly

<http://www.websamba.com/ldulman/vadaonix.zip>

Leonid Dulman

[See also "VAD - Visual Ada Developer" in AUJ 23-3 (Sep 2002), p.134. -- dc]

Ada for E-mail Servers or Clients

*From: Pascal Obry <p.obry@wanadoo.fr>
Date: 18 Nov 2002 17:27:25 +0100
Subject: Re: Ada Mail Server?
Newsgroups: comp.lang.ada*

> Is there a mail server for Windows programmed in Ada?

Depending on what you are looking for! In my SMTP package I have a part for the server. This is more a demo than a full server ;) all it does is read incoming SMTP mail, parse them and create an Ada structure with this information and call a callback routine for each message... This was to implement some kind of mail-server (sending command in e-mail to activate such or such command...)... Anyway all this is work in progress... You can grab current sources from my homepage. Look for package SMTP.

[See

<http://perso.wanadoo.fr/pascal.obry/contrib.html>
-- dc]

Pascal Obry, Team-Ada Member, Magny Les Hameaux, France

*From: David C. Hoos
<david.c.hoos.sr@ada95.com>
Date: Mon, 18 Nov 2002 15:31:24 -0600
Subject: Re: Ada Mail Server?
Newsgroups: comp.lang.ada*

> I was hoping for a full service mail server that would handle sending and receiving (from multiple ISPs) for all users/computers on a home network. I have downloaded your SMTP package and will take a look at it. [...]

Do you really mean a server, or are you talking about an E-mail `_client_`? If you mean a `_server_`, are you planning to open

port 25 to the world on your home network?

When you talk about multiple ISPs, that sounds like you want an E-mail client, because ISPs usually provide an SMTP server accepting connections on port 25, and a POP3 server accepting connections on port 110.

I have used Pascal's SMTP package, but what it does (in its default configuration) is listen on port 25 for incoming messages which have been addressed to the host on which the package is running -- i.e., the messages were `_not_` addressed to some user@ some ISP.

I used Pascal's package integrated into a program handling tactical text messages using SMTP as the transport. This way, my application reacted immediately to an incoming message, rather than depending on some program like sendmail signalling my application.

So, all that said... it doesn't sound like what you want.

From: Alfred Hilscher <Alfred.Hilscher@t-online.de>

Date: Fri, 22 Nov 2002 18:09:32 +0100

Subject: Re: Ada Mail Server?

Newsgroups: comp.lang.ada

> On a related subject, is there a mail client (eg POP3) for Windows programmed in Ada?

Not a client, but a set of POP3-routines:

<http://alfred.hilscher.bei.t-online.de/sophi.zip>

Bindings to File and Directory Operations

From: Preben Randhol <randhol@pvv.org>

Date: Mon, 11 Nov 2002 09:45:02 +0000 (UTC)

Organization: Norwegian university of science and technology

Subject: Re: "directories" help

Newsgroups: comp.lang.ada

> I want to write Win32 program in Ada using GNAT 3.13 [...] I want to work with "folders" (list content, look for files, search them, look attributes), and I want it to work also on Unix-like OS like FreeBSD or Linuxes.

GNAT.OS_Lib =>

http://www.adapower.com/lang/gnatrun.dead/gnat-os_lib.html

Overview =>

<http://www.adapower.com/lang/gnatrun.dead/packages.html>

Examples =>

<http://www.adapower.com/lang/>

From: gdemont@hotmail.com (Gautier)

Date: 11 Nov 2002 09:08:41 -0800

Subject: Re: "directories" help

Newsgroups: comp.lang.ada

Fortunately there is GNAT.OS_Lib and others that are fully portable. A few examples in "gnatpaqs.zip" on page below.

<http://www.mysunrise.ch/users/gdm/gsoft.htm>

From: Jacob Sparre Andersen

<sparre@nbi.dk>

Date: Tue, 14 Jan 2003 22:41:08 +0100

Subject: Re: Directory Listing

Newsgroups: comp.lang.ada

If you want a more general solution [than GNAT.OS_Lib -- dc], you should use a POSIX library (Florist is one option).

From: Pascal Obry <p.obry@wanadoo.fr>

Date: 11 Nov 2002 16:49:08 +0100

Subject: Re: "directories" help

Newsgroups: comp.lang.ada

First upgrade to GNAT 3.14p and use GNAT.Directory_Operations... And be sure to read the GNAT Reference Manual, there is a special section listing all nice packages provided with GNAT!

From: R. Srinivasan

<r.srinivasan@cox.net>

Date: Fri, 17 Jan 2003 02:28:35 GMT

Subject: Re: Directory Listing

Newsgroups: comp.lang.ada

Apart from all the splendid ideas, I suggest you also take a look at the package Gnat.Command_Line and in particular procedures related to filename expansion.

If you need an example, I have a sample program "findfiles" that is a cheap find (the Unix util) in Ada heavily relying on gnat packages mentioned - please contact me by email.

Towards a Standard Ada OS Binding

From: Jean-Pierre Rosen

<rosen@adalog.fr>

Date: Thu, 14 Nov 2002 18:14:39 +0100

Subject: Ada OS binding

Newsgroups: comp.lang.ada

There has been a lot of discussion, as well as amendments, about adding packages that provide access to OS services, such as browsing a directory, getting information about files, and (a lot) more.

Such a proposal needs to be discussed with the community-at-large, but it would be inappropriate to load cla with such discussions. I have therefore opened a page where you can get information about the current status at

http://www.adalog.fr/os_bind/

and I have established a mailing list:

http://groups.yahoo.com/group/ada_osbind/

If you are interested in this effort, please visit these pages, join in, and with your help we may hopefully achieve something that would be acceptable for AdaOY!

From: Thomas

<fantome.forums.deContes@iFrance.com>

Date: 14 Jan 2003 00:46:12 GMT

Subject: os services

Newsgroups: fr.comp.lang.ada

[Extracts translated from French: -- dc]

I suggested the OS services of J-P. Rosen to several people:

http://www.adalog.fr/compo/os_services.htm

[See also "OS File System Bindings" in AUJ 23-1 (Mar 2002), pp.13-14. -- dc]

RTDB & FIRM - Real-Time Database Products in Ada

From: Wes Groleau

<wesgroleau@despammed.com>

Date: Fri, 13 Dec 2002 09:13:05 -0500

Organization: Raytheon Company

Subject: Re: IBM Acquires Rational Ada

Newsgroups: comp.lang.ada

> So, if someone were to create a database product in Ada for Ada, would there be a market for it? [...]

RTDB -> FIRM. Developed under US DoD contract, therefore source can be acquired from the DoD. Being DoD, however, the red tape to get it would be a pain.

For less hassle (but more money), buy them from Lockheed-Martin with support. At least I think you can. [...]

[And from a later message: -- dc]

> [...]. I agree that it would be *really* slick if Ada compilers were to come with some sort of database available. But how do we get the money to build something like that? The government? But they don't want to fund Ada development, do they? [...]

They already did. See my answer [above]. I forgot to mention RTDB was Ada 83 and was written because Ingres was way too slow for AN/BSY-2. FIRM was its Ada 95 successor (sort of).

Note that both were developed under defense contracts, with all the requirements, design, test, peer review, etc. rigor which that implies.

From: Wes Groleau

<wesgroleau@despammed.com>

Date: Mon, 16 Dec 2002 08:56:02 -0500

Organization: Raytheon Company

Subject: Re: IBM Acquires Rational Ada

Newsgroups: comp.lang.ada

> Sounds like a good start. [...] Is any of that stuff available under any kind of license that lets someone else use/develop it?

It is my belief that since we (the taxpayers) paid for it, we (the taxpayers) can get the source, provided it isn't classified -- which it shouldn't be, being a general-purpose database. [...]

> Is it sufficiently powerful - or at least offering something new/unique - that it could compete against other database products?

Seemed like it to me. But contact Lockheed-Martin, Box 4840, Syracuse, NY for the details. They do not seem to be aggressively marketing it. I did not work directly on RTDB and I left GE/Martin/Lockheed before the successor, FIRM, came along. But I liked what I read about both. [...]

BTW, RTDB = Real-Time DataBase. A fast_and_deterministic component for a hard-real-time_large_military system.

[...] you did suggest developing a database, and I figured mentioning the existence of a good one couldn't hurt.

*From: faust <urfaust@optushome.com.au>
Date: Sat, 21 Dec 2002 18:41:36 -0800
Subject: Re: IBM Acquires Rational Ada
Newsgroups: comp.lang.ada*

[...] see

http://www.lockheedmartin.com/syracuse/eaglesped/products/rtdb/rtdb_features.html

[And from a later message: -- dc]

> For less hassle (but more money), buy them from Lockheed-Martin with support. At least I think you can.

You can. See

<http://www.lockheedmartin.com/syracuse/eaglesped/products/firm/firm.html>

GNAT Ada 95 Database Development Environment

*From: Michael Erdmann
<Michael.Erdmann@snaflu.de>
Date: Sun, 29 Dec 2002 10:47:38 +0100
Subject: Release of GNAT Ada Database Development Environment 1.4.1
Newsgroups: comp.lang.ada*

[See also same topic in AUJ 23-4 (Dec 2002), p.197. -- dc]

The Version 1.4.1. of the GNADE project available at:

<http://gnade.sourceforge.net/> and
<http://sourceforge.net/projects/gnade>

for Linux, Windows and Solaris 2.8.

This environment allows, using the GNAT Ada 95 compiler, to implement Ada 95 applications invoking relational database products such as MySQL, PostgreSQL, MimerSQL and Oracle.

This release includes besides of bug fixes the following features: thin bindings to the ODBC interface; ISO 92 embedded SQL preprocessor for Ada 95 generating code for the ODBC interface; Oracle Call Interface; bindings for MySQL; bindings for PostgreSQL; draft implementation of the ADBC interface; documentation in pdf, postscript and html format.

The release if available as source code and prebuild rpm files for Linux.

Chess Program in Ada

*From: Mårten Woxberg
<marwo264@student.liu.se>
Date: Mon, 04 Nov 2002 16:49:24 +0100
Subject: Re: Game algorithm
Newsgroups: comp.lang.ada*

Me and my friends did a Chess program in Ada for a school project. We used a weighted minmax algorithm for our two AI-clients. It was a client server approach since we needed networking for the project. Our server could handle many games at the same time (tested with 4Gb RAM so we didnt care). AI-clients ate through 60.000 boards before doing a move. that's about 4 levels deep into the move tree.

I could probaly get you the whole source (client, server and ai). We used a local package for drawing colors in Sun X-terms so that probably won't work well.. there should be a quick 'n dirty version of it too...

If you want it just say so...

Idea for Game in Ada

*From: Steven Deller <deller@smsail.com>
Date: Mon, 4 Nov 2002 20:13:20 -0600
Organization: Smooth Sailing LLC
Subject: RE: Game algorithm
Newsgroups: comp.lang.ada*

> One of the "future" projects I am planning is a chess program in Ada.

Instead of chess, you might want to look into "Go". The rules are simpler than chess but the complexity is far greater (huge orders of magnitude greater). And it requires as much "feel" as "analysis".

Plus there is a \$1,000,000 prize waiting for the first program to beat a Pro player at the lowest level. So far, no program has even gotten close. In fact, I am a good (not great) amateur player, and I have no problem giving any Go program a handicap and still beating it without difficulty. If this piques your interest, check out: <http://www.usgo.org>

AdaSDL Binding to Multiplatform Game Development Library

*From: Stefan Soos <stefan.soos@gmx.de>
Date: Fri, 1 Nov 2002 22:29:26 +0100
Subject: Re: OpenGL and Glut bindings
Newsgroups: comp.lang.ada*

> I've been looking around for bindings to OpenGL or Glut.

You could try the thin binding to the SDL library. SDL itself is a multiplatform game development library with support for OpenGL and Glut. You can find SDL at <http://www.libsdl.org> and the Ada binding at <http://adasdl.sourceforge.net>.

[See also "AdaSDL - Binding to Simple DirectMedia Layer (SDL)" in AUJ 23-3 (Sep 2002), pp.133-134. -- dc]

Toy Breakout game in Ada

*From: Chad R. Meiners
<crmeiners@hotmail.com>
Date: Mon, 2 Dec 2002 19:58:58 -0500
Organization: Michigan State University
Subject: A toy breakout game in Ada
Newsgroups: comp.lang.ada*

If anyone is interested, I have posted a simple and partly complete breakout game that uses the SDL libraries. It is at

<http://www.cse.msu.edu/~meinersc/bricks/abricks.zip>

It includes a windows binary and source code. Be warned that the source code was cobbled together to test the functionality of the AdaSDL thick binding. When time permits I may extend and clean up the source code in the future. Feel free to send feedback if you enjoy the program.

*From: Chad R. Meiners
<crmeiners@hotmail.com>
Date: Wed, 8 Jan 2003 22:46:16 -0500
Organization: Michigan State University
Subject: Feedback wanted: Breakout game using SDL
Newsgroups: comp.lang.ada*

At [URL above] is a breakout game I wrote in Ada using the AdaSDL thick binding. I fortunately had some spare time to clean up the code for the game, and I am making the new and improved code (and a w32 binary) available under the GPL. I would like the following feedback if any of you have the time.

- (1) Performance: How does the game perform on your computer?
- (2) Portability: I only have access to win2K machines. If someone is willing to try and build the program on Linux or Mac, I would welcome feedback on the results.
- (3) Thick Binding Clarity: The purpose of writing to game was to test how easy the AdaSDL thick binding is to use. Does the binding look convenient and easy to use?

The game is in a basic stage. You do not have to worry about losing the game and you knock bricks down until you feel like quitting. In time, I plan on adding bells and whistles to the game.

*From: Chad R. Meiners
<crmeiners@hotmail.com>
Date: Tue, 4 Feb 2003 19:24:25 -0500
Organization: Michigan State University
Subject: ANN: Breakout game Updated
Newsgroups: comp.lang.ada*

<http://www.cse.msu.edu/~meinersc/bricks/abricks.zip>

Has been updated to make the game more user friendly: (1) No longer runs it fullscreen. (2) Releases the mouse when

paused. Has anyone gotten the code to compile and run on a unix system?

Multi-User Dungeon (MUD) Game in Ada

From: David95038@aol.com (David Kristola)
Date: 3 Dec 2002 03:44:20 -0800
Subject: Style question: deep inheritance
Newsgroups: comp.lang.ada

I am working on a MUD in Ada. [...]
 [See <http://members.aol.com/drveg/mud/> -- dc]

Ada Parsing for Tool Builders

From: daniel gaudry
<Daniel.Gaudry@wanadoo.fr>
Date: Mon, 11 Nov 2002 00:24:04 +0100
Subject: Re: Lex et yacc pour ada
Newsgroups: fr.comp.lang.ada

[Extracts translated from French: -- dc]
 > Where can one find a lex and a yacc (or equivalent) for Ada? The goal would be to produce a static analyzer for code [...]

A complete Ada parser with code in Ada is built-in in the editor (code available) at <http://www.dgaudry.com/>. A separate version is available on request.

From: Jean-Pierre Rosen
<rosen@adalog.fr>
Date: Tue, 12 Nov 2002 13:23:02 +0100
Subject: Re: Lex et yacc pour ada
Newsgroups: fr.comp.lang.ada

The parser is only a very small part of a non-trivial static analyzer. Best for this kind of things is to use ASIS, which provides all semantic information.

Adasubst & Adadep - Semantic Tools for Ada

From: Jean-Pierre Rosen
<rosen@adalog.fr>
Date: Thu, 21 Nov 2002 16:22:05 +0100
Subject: New version of semtools (Adasubst/Adadep) released
Newsgroups: comp.lang.ada

Adalog is pleased to announce version 1.1 of Adasubst/Adadep, the tools that help you reorganize your projects! As usual, it is freely available from Adalog's components page:

<http://www.adalog.fr/compo2.htm>

[See also "Adasubst & Adadep - New Free Semantic Tools for Ada" in AUJ 23-3 (Sep 2002), p.126. -- dc]

New in this version of both utilities:

Use GMGPL license. Does not change much in practice, but it seems that company lawyers hardly start to understand the GPL, so it's better not to invent new licenses. Note that this license allows you not only to use, modify, etc. the software, but also to reuse any part of

it in your own (possibly proprietary) applications.

New option -p: get the "-I" options from an ".adp" file

New in this version of Adasubst:

New syntax of the dictionary. Presumably (much) easier to write and understand. Allows exceptions to substitutions (i.e. definition of elements that are *not* to be substituted).

Support for derived type: if you substitute the name of a primitive operation of a type, the substitution is propagated to all derived types (unless there is an explicit substitution for the derived operation, of course).

Supports substitution of operators. And if you change an operator to a regular function, all infix calls are changed into prefixed calls.

Support for overloaded entities. You can have different substitutions for overloaded entities with the same name.

-l option allows to limit the output line length (useful in cases where the substitution makes lines longer than what is acceptable to the compiler).

If the output name is specified with -o, output files who suffered no substitution are deleted (so only modified files are produced). -O option keeps all files.

Many bug fixes, improvements to the documentation, etc.

Tools to Convert C to Ada

From: Ira Baxter
<idbaxter@semdesigns.com>
Date: Mon, 25 Nov 2002 09:10:11 -0600
Subject: Re: C to Ada for Windows?
Newsgroups: comp.lang.ada

> [Is there] a program that converts C to Ada for Windows OS? Thanx in advance!

I don't know of a tool that specifically does this.

We offer generalized compiler technology for carrying out such conversions, however, and our tools have a full ANSI C front end complete with preprocessor, and an Ada back end. Somebody has to code the actual transforms. See

<http://www.semdesigns.com/Products/Services/Le-gacyMigration.html>.

I'm not sure how pretty the results will be. C programs don't have a lot of explicit modularity or careful type declarations ("int X" doesn't tell you much about the expected range of values for X). One could, with effort, extract some of that range information by automating a program analysis.

Ira D. Baxter, Semantic Designs, Inc.

[See also "Semantic Designs - Custom Translation Tools" in AUJ 23-4 (Dec 2002), pp.203-204. -- dc]

From: jorg.van.daelen@zonnet.nl (Jorg van Daelen)

Date: 27 Nov 2002 11:07:35 -0800
Subject: Re: C to Ada for Windows?
Newsgroups: comp.lang.ada

I tried for some time to port c2ada to Windows but there are too many hurdles. It takes less effort to install Redhat Linux and use the pre-build c2ada converter from home.pacbell.net/nma123

[Dmitriy Anisimkov
 <anisimkov@yahoo.com> added: -- dc]

CBIND is converting at least C header files to Ada. There is a port to Win32.

<http://vagul.tripod.com/cbind4win32.zip>

[See also "Tools for Creating Ada Bindings to C Headers" in AUJ 22-4 (Dec 2001), p.202. -- dc]

From: Craig Carey <research@ijs.co.nz>
Date: Fri, 29 Nov 2002 19:04:07 +1300
Subject: Re: C to Ada for Windows?
Newsgroups: comp.lang.ada

Sure there could be a C to Ada parser.

ASF+SDF seems to be a best/quality choice for implementing a C to Ada 95 parser. It can even separate out "xy" as "x * y" and implement the mixfix notation. [...]

Here is the ASF+SDF reference manual:

<http://www.cwi.nl/projects/MetaEnv/meta/doc/manual/user-manual.html>

[...]

Improved OpenToken Version

From: francois_fabien@hotmail.com
Date: 28 Nov 2002 22:15:58 -0800
Subject: Evolution d'OpenToken
Newsgroups: fr.comp.lang.ada

[Translated from French: -- dc]

An improved version of Ted Dennison's OpenToken package with a corrected LALR parser and a syntax tree builder is downloadable from

<http://firfabien.chez.tiscali.fr/>

[See also "OpenToken 3.0b - Token Analysis and Parsing Package" in AUJ 21-3 (Oct 2000), pp.165-166. -- dc]

P2Ada - Pascal to Ada Translator

From: Gautier de Montmollin
<gdemont@hotmail.com>
Date: Sat, 11 Jan 2003 19:36:10 +0000
Subject: Ann: P2Ada (v.10-Jan-2003)
Newsgroups: comp.lang.ada

Hello - there is some progress on P2Ada, the command-line Pascal to Ada translator. There is now a preprocessor (BP2P) that transform a few low-level

details, mainly of Borland Pascal before the translation by P2Ada.

URL:

<http://www.mysunrise.ch/users/gdm/gsoft.htm#p2ada> [...]

[See also "P2Ada - Turbo/Borland Pascal to Ada Translator" in AUJ 20-4 (Jan 2000), p.237. -- dc]

From: Gautier de Montmollin
<gdemont@hotmail.com>

Date: Fri, 31 Jan 2003 23:44:45 +0000

Subject: Ann: P2Ada (v.31-Jan-2003)

Newsgroups: comp.lang.ada

Again a bit further on the side of Pascal to Ada translation...

P2Ada now comprises the memorization engine of a Pascal compiler. Roughly, this has two important consequences: there can be a lot of predefined items (types, functions) with their Ada translation, naturally masked when they are [re]defined in the Pascal source; the WITH instruction is correctly translated.

BP2P, the preprocessor, understands conditional defines (symbols can be set) and makes use of the \$B+/\$B- switch states (Boolean evaluation).

Other smaller improvements in the archive's "whatsnew.txt". [...]

Platform Independent Access to Configuration Information

From: Stephen Leake

<stephen.a.leake.1@gsfc.nasa.gov>

Date: 31 Oct 2002 12:22:37 -0500

Organization: NASA Goddard Space Flight Center (skates.gsfc.nasa.gov)

Subject: Re: problem with abstract types

Newsgroups: comp.lang.ada

> [...] I want to write an OS independent way to store configuration informations (on Windows in the registry and on Unix in human readable config files).

I am hurt, shocked, and amazed that noone has pointed you to Grace, at

<http://savannah.nongnu.org/projects/grace/>

It currently has a draft version of the Config_Files package, which does exactly what you want, and was discussed extensively last spring.

Unfortunately, the administration of the Grace web site is a bit lacking (hint to Ted Dennison :), so the easiest way to get the Config_Files package is by browsing the CVS repository, at:

<http://savannah.nongnu.org/cgi-bin/viewcvs/grace/grace/> and the

documentation for Config_Files is only available as TeXinfo source, at:

<http://savannah.nongnu.org/cgi-bin/viewcvs/grace/doc/>

I hope to make this cleaner soon. [...]

From: Stephen Leake

<stephen.a.leake.1@gsfc.nasa.gov>

Date: 01 Nov 2002 12:21:31 -0500

Organization: NASA Goddard Space Flight Center (skates.gsfc.nasa.gov)

Subject: Re: problem with abstract types

Newsgroups: comp.lang.ada

> [...] an ANNOUNCE would have been nice (perhaps I have missed it).

Well, my phrasing was mostly tongue-in-cheek (I guess I should have worked in a smiley somewhere). There has not been an explicit announcement, because of my problems with the Grace web site. Ted has agreed to make me an administrator, so things should get better soon.

Be aware that the code in Grace.Config_Files does `_not_` currently meet the full requirements for Config_Files spelled out in the document. Feel free to work on it; I can make you a Grace developer if you'd like.

From: dennison@telepath.com (Ted Dennison)

Date: 4 Nov 2002 06:30:11 -0800

Subject: Re: problem with abstract types

Newsgroups: comp.lang.ada

I'd also like to point out that there are thick Win32 registry bindings in the SETI@Home Service, for which sources are available. The sources are GPL, but I've always intended to release the bindings into the public domain, or find them a good home elsewhere. So if anyone is interested in them but put off by the GPL, contact me.

[See

http://www.telepath.com/~dennison/Ted/SETI/SETI_Service.html and also "SETI@Home Service 2.1" in AUJ 22-3 (Sep 2001), p.148. -- dc]

From: Ulrich Eckhardt <uli.e@gmx.de>

Date: Tue, 03 Dec 2002 17:47:24 +0100

Subject: Ada Configuration API for Unix and Windows

Newsgroups: comp.lang.ada

As promised some time ago, I have finished my platform independent API for accessing configuration informations (Copyright GPL). More infos on http://www.uli-eckhardt.de/ada_registry/. Comments and suggestion are welcome.

Retrieving Output from Spawned Processes

From: Marc A. Criley

<mcq95@earthlink.net>

Date: Thu, 07 Nov 2002 01:48:15 GMT

Subject: Re: retrieve command result

Newsgroups: comp.lang.ada

> Is it possible to retrieve the result of a command in a variable?

You are looking for "Pipes", which is an excellent piece of software written by Jim Rogers. It makes doing things like this very easy.

<http://www.adapower.com/reuse/pipes.html>

Marc A. Criley, Quadrus Corporation, www.quadruscorp.com

From: Emmanuel Briot <briot@act-europe.fr>

Date: 07 Nov 2002 10:43:14 +0100

Subject: Re: retrieve command result

Newsgroups: comp.lang.ada

There is a package, which [...] was already there with 3.14p, called GNAT.Expect which can be used to do this sort of things. It is more aimed at interactive programs, where you want to keep sending input and receiving output from the second process, but it might be used in the simple scenario you want as well.

This is portable to most systems where GNAT is ported (apart from VMS I believe), and is the package used by GVD to communicate with gdb.

Rot-13 Encryption

From: Preben Randhol

<randhol@pvv.org>

Date: Thu, 7 Nov 2002 15:15:18 +0000

(UTC)

Organization: Norwegian university of science and technology

Subject: Re: Rot-13 (caesar code)

Newsgroups: comp.lang.ada

Rot-13 (short for rotate 13) is a simple letter substitution encryption scheme. It works by replacing the current letters in a message with those that are 13 positions ahead in the alphabet. The letter a is replaced by n, b by o, c by p, etc. Numbers and punctuation are not encoded.

Now look at my simple program:

<http://www.pvv.org/~randhol/Ada95/ainvisiclude/ainvisiclude.html>

You can easily change this to deal with Rot-13.

From: Adrian Knoth

<adi@drcomp.erfurt.thur.de>

Date: 7 Nov 2002 21:16:18 GMT

Organization: Modern Electronics

Subject: Re: Rot-13 (caesar code)

Newsgroups: comp.lang.ada

<http://adi.thur.de/?show=rot13>

FSMedit - Finite State Machines Editor

From: Christoph Grein

<christoph.grein@eurocopter.com>

Date: Mon, 11 Nov 2002 06:27:59 +0100

(MET)

Subject: FSMedit 3.0

Newsgroups: comp.lang.ada

FSMedit - The Finite State Machines Editor - Version 3.0

Now, additionally to Claw, FSMedit is also available with GtkAda, so it's no longer restricted to Windows.

[See also same topic in AUJ 23-1 (Mar 2002), pp.16-18. -- dc]

FSMedit aids in specifying transition diagrams for Finite State Machines. When the transition diagram of a FSM has been specified, a simulation can be executed interactively or via scripts. From the transition diagrams, Ada code can be generated; also the reverse, extraction of transition diagrams from Ada code is possible. Tools for reachability analysis and for comparison of machines are also available.

FSMedit may be installed in different languages. Currently English and German are selectable. Further languages can be provided without recompilation, solely by providing the translation in a language file.

FSMedit has been released under the Gnu GPL. However, since it is an "editor", any software (i.e. Finite State Machine) you produce with it does not fall under any licence other than you choose yourself.

To download FSMedit together with Ada source code and a complete user's manual, see: <http://home.t-online.de/home/Christ-Usch.Grein/Ada/FSM.html>

From: Christoph Grein
<christoph.grein@eurocopter.com>
Date: Wed, 29 Jan 2003 08:11:49 +0100
(MET)
Subject: FSMedit V3.1
Newsgroups: comp.lang.ada

FSMedit - The Finite State Machines Editor - Version 3.1

New in this version: Undo/Redo facility and tooltips. [...]

Christoph Grein, Member of Ada Germany, <http://www.ada-deutschland.de>,
 Christ-Usch.Grein@T-Online.de

Physical Units in Ada - a Survey

From: Grein, Christoph
<christoph.grein@eurocopter.com>
Date: Mon, 11 Nov 2002 06:35:57 +0100
(MET)
Subject: Announce: Survey over Handling
Units of Measure (Physical Dimensions)
in Ada
Newsgroups: comp.lang.ada

There is a new page dedicated to physical units in Ada. There are several methods how to compute with guaranteed dimensional correctness. This page presents four of them so that you can choose the optimal method for your application at hand. See

<http://home.T-Online.de/home/Christ-Usch.Grein/Ada/Dimension.html>

Ada Mode for JED Editor

From: kees.serier@ordina.nl (Kees Serier)
Date: 12 Nov 2002 23:59:48 -0800

Subject: Re: Alternatives to Emacs under Linux?

Newsgroups: comp.lang.ada

> What are freeware alternatives to Emacs (and Vi) as Ada IDE under Linux? [...]

You might try JED, an editor with syntax highlighting for many languages, multiwindow, C-like macro language called 'S-Lang', a very good emacs mode, free software with source, runs on UNIX (character mode and X) Windows and VMS. <http://www.jedsoft.org/jed/> [...]

The Ada mode (from Lutz Donnerhacke) is not include with JED, but can be found at:

<ftp://ftp.iks-jena.de/mitarb/lutz/ada/ada.sl>.

You must first include it in site.sl before you can use it.

Geodetic Functions

From: Fred Killet <killet@killetsoft.de>
Date: Sun, 12 Jan 2003 14:08:24 +0100
Subject: GEODLL - Geodetic functions
Newsgroups: comp.lang.ada

GEODLL - Geodetic functions - New version

A new version 8.00 of GEODLL32, a DLL containing a lot of geodetic functions, is ready for a download from the site <http://www.killetsoft.de/geodlle.htm>. GEODLL32 is build as a "foreign host dll" and so it can be used from most programing languages like C++, Delphi, Access, CA-Visual Objects, Visual Basics and so on.

In this time the DLL is documented in German languages. But the functions headers are mostly self explaining and so you should not have any problems to implement the functions in your own applications. I hope to provide an English language version by the end of the year. [...]

From: Eric G. Miller <egm2@jps.net>
Date: Sun, 12 Jan 2003 16:39:52 GMT
Subject: Re: GEODLL - Geodetic functions
Newsgroups: comp.lang.ada

Interesting. I have a thin binding to the U.S. NIMA's geotrans2.2.1 library posted at <http://pweb.jps.net/~egm2/libgeotrans/index.html> ["a datum and coordinate conversion library" -- dc]. I've been meaning to do a thick binding but haven't gotten around to it yet... The library already has C, C++ and Java-JNI interfaces (it's written in C).

ZLib.Ada - Thick Binding to ZLib Compression Library

From: Anisimkov <anisimkov@yahoo.com>
Date: Mon, 20 Jan 2003 08:08:05 +0600
Subject: An: ZLib.Ada
Newsgroups: comp.lang.ada

<http://zlib-ada.sourceforge.net/>

ZLib for Ada thick binding (ZLib.Ada). Release 1.0

ZLib.Ada is a thick binding to the popular compression/decompression library ZLib <http://www.gzip.org/zlib/>. ZLib.Ada provide Ada style access to the ZLib C library.

Ada-related Products

ACT - New Public Release GNAT 3.15p

From: Arnaud Charlet
<charlet@gnat.com>
Date: Mon, 18 Nov 2002 18:31:23 +0100
Subject: ANNOUNCE: GNAT 3.15p release
Newsgroups: comp.lang.ada

GNAT is a complete Ada95 compilation system, maintained and distributed under the GNU Public License by Ada Core Technologies.

GNAT is based on GCC, the GNU Compiler Collection developed by the Free Software Foundation.

From time to time, Ada Core Technologies makes available public versions of the GNAT technology, derived from the same code base as the commercial GNAT Pro technology. These versions are freely and widely downloadable from multiple web sites at no charge. They are intended for student and research use, where a comprehensive Ada 95 compiler technology is needed, but no support of any kind is required.

This is NOT an evaluation version of GNAT Pro: if you want to evaluate an industrial-strength, fully supported Ada compiler, please contact either Ada Core Technologies (sales@gnat.com) or ACT Europe (sales@act-europe.fr) for information on availability and support services for the GNAT Professional edition.

The current distribution of the public version of GNAT is 3.15p. This version is based on GCC 2.8.1. It should not be confused with the version of GNAT available with GCC 3.2, which is an alpha version of the integration of GNAT with recent versions of the GCC code generator, and does not correspond to any specific GNAT release.

GNAT 3.15p is available at <http://libre.act-europe.fr/GNAT> and is mirrored at <ftp.cs.nyu.edu/pub/gnat>

The list of new features is available at <http://libre.act-europe.fr/GNAT/3.15p/features>
 Arnaud Charlet, ACT Europe

Aonix - Release of ObjectAda for Linux

URL:
http://www.aonix.com/content/news/pr_1.15.02.html

Press Releases

Aonix® Announces the Release of ObjectAda® for Linux (Intel®/Linux)

San Diego, California, November 15, 2002 Aonix, a member of the Gores Technology Group and a leading provider of Ada 95 software development environments, is pleased to announce that ObjectAda® for Linux (Intel®/Linux), the object-oriented development environment, is now available.

Aonix, the leading supplier of quality Ada technology, is releasing ObjectAda for Linux as the first in a series of ObjectAda product releases designated as version 8.0. Version 8.0 ObjectAda product technology passes the latest Ada Conformity Assessment Test Suite (ACATS), version 2.5. ACATS is the most comprehensive Ada 95 validation test suite available today and ensures high quality for this latest development environment.

The ObjectAda for Linux product ships with all the popular ObjectAda standard features. ObjectAda for Linux comes with both a graphical and command line interface; integrated language-sensitive editor; lightweight source-based library model; and lightning-fast compilation. The ObjectAda for Linux compilation system is composed of the compiler, debugger, browser, editor, and full library manager.

In addition to the basic compiler development package, the more advanced package, Project Pack, contains the Ada-ASSURED advanced editor that provides additional language-sensitive features and style-guideline conformance checking. Project Pack also contains the AdaNav toolset, which provides complete system HTML source-navigation capabilities as well as call tree and unit tree graphical reporting and automatic data dictionary generation. The AdaNav profiler provides run-time performance reporting to help developers identify application hot spots to improve program performance.

ObjectAda for Linux is rigorously tested on the Red Hat Linux operating system. And it will also run on a variety of other Intel-based Linux operating systems, states Greg Gicca, Product Manager for ObjectAda. Thus, this release provides support for a whole new set of possible host development platforms. [...]

Aonix, a Gores Technology Group company, is a leading international software company with customers drawn from the Global 1000. The Critical Development Solutions (CDS) division produces Software through Pictures® (StP), Architecture Component Development (ACD), TeleUSE®, ObjectAda®, AdaWorld® and Raven. CDS products support the highest criticality levels of software design. [...] Aonix operates sales offices throughout

North America and Europe in addition to a network of international distributors. For more information, visit www.aonix.com. [...]

Aonix - Commitment to Ada

*From: David Humphris
<dhumphris@aonix.co.uk>
Date: Wed, 15 Jan 2003 09:04:03 +0000
Subject: Re: Aonix commitment to Ada
Newsgroups: comp.lang.ada*

[...] We have some very good news! First, Ada has been doing pretty well within Aonix in recent years even if [...] the main focus of the company had not always been on Ada.

The good news is that the old Aonix just split 2 weeks ago into two new companies: the new Aonix, which retains the modeling (StP/ACD), Ada and UIMS (TeleUSE) tools businesses and Select Business Solutions, which consists of the former eBIS and Select divisions.

We can officially announce that the new Aonix is committed to keep its Ada business healthy and growing. The new investment structure will allow Aonix to refocus on its typical mission-critical applications business, where Ada is playing a key role, especially in the safety critical area (but not only!).

It is always easy to make announcements. It is sometimes more difficult to show some achievements. That's why we invite all the Ada users to stay tuned to see what Aonix is going to deliver as new features and products in the Ada world very soon.

David Humphris, Real-Time Sales Consultant, Aonix

Aonix - New ObjectAda Special Edition Forthcoming

*From: rod.chapman@praxis-cs.co.uk (Rod Chapman)
Date: 19 Jan 2003 06:18:40 -0800
Subject: Re: Aonix commitment to Ada
Newsgroups: comp.lang.ada*

I'm pleased to say that we hope to include an up-to-date version of the Aonix ObjectAda Special Edition for Windows on the CDROM that will accompany the new edition of "High Integrity Software: The SPARK Approach" by John Barnes, to be published soon by Addison Wesley. Thanks go to David and his colleagues at Aonix for supplying this.

(The CDROM will also include the SPARK toolset, of course! Oh... and GNAT 3.15p for Windows and Linux as well... :-))

Aonix - Updated AdaJNI Release

*From: owner-intel-objectada
Date: Thu, 30 Jan 2003 09:54:18 -0800
Subject: Intel-OA: New updates available*

To: intel-objectada@aonix.com

A new patch update 1102V722-U7 to ObjectAda for Windows 7.2.2 (1102) is now available. In addition, a new add-on update 1102V72x-A1 for Enterprise Edition customers supplies the latest AdaJNI release.

The update download files and the Release Notes are available at <http://www.aonix.com/content/support/ada/patches/objectada.html>

Please see the Release Notes for more information.

Aonix Ada Support, adasupport@aonix.com, 1-800-972-6649

Artisan - Real-time Studio's Ada Support

*From: alveryg@artisansw.com (Alvery Grazebrook)
Date: 12 Dec 2002 06:41:54 -0800
Subject: Re: IBM Acquires Rational Ada
Newsgroups: comp.lang.ada*

> Artisan has Ada support, and is fairly well-known to real-time users. As far as I can tell, it's rather fixed in its views; [...]

The Artisan tool, Real-time Studio has been evolving its Ada code support. The code-generator is template based, so you can customize it any way you like. It also includes reverse engineering and what we call "Synchronization". This is basically a differencing engine to compare the current state of the code with the current state of the model, combined with a resolution capability that will re-generate or reverse any parts that you select based on the differences.

Give it a try if you want. You can download a copy from

http://www.artisansw.com/eval/eval_download.asp

Alvery Grazebrook (Product Manager, Real-time Studio)

*From: alveryg@artisansw.com (Alvery Grazebrook)
Date: 16 Dec 2002 05:24:51 -0800
Subject: UML to Ada Mapping
Newsgroups: comp.lang.ada*

> But is reversing and synchronisation available once you customise the templates? This isn't just a question for Artisan but for all UML tools.

Since you ask, there are 2 parts to the answer.

The simple answer is that the reversing works to a defined mapping, and the synchronizer supports a pre-defined range of possible mappings when producing the comparison view. This means that synchronization works cleanly for all generation templates that generate code that conform to the semantics of the

reverser's Ada -> UML mapping, and will show differences otherwise.

The second part to the answer is that the job of a synchronizer is to help translate UML model to Ada and back. Given that the semantics of Ada is well defined, and the semantics of UML is reasonably well defined, the only part of the mapping that is at issue is the way you need to annotate the UML model to cover the features supported by Ada that aren't supported by UML. The Ada UK community has taken up the challenge of developing an Ada Profile (for UML) to define the UML extensions required for the mapping.

A profile in UML is a set of extended properties that can be attached to UML elements (e.g. operations, classes) to identify them as having a particular significance.

For example, Ada 95 has several "structures" that can contain variables, sub-programs etc. they are: package, task type, protected type, and subprogram. UML has only two: class, package. Therefore the UML Class, in particular, needs extra annotation to determine whether it should map to an Ada package, a task type or a protected type.

Does anyone in this community have an interest in participating in the definition of an Ada profile?

Alvery (Product Manager, Artisan software tools, www.artisansw.com)

DDC-I - Joins Forces with BAE SYSTEMS on ARINC-653 Operating System

URL:

http://www.ddci.com/news_bae_ddci_join_release.shtml

BAE SYSTEMS & DDC-I Join Forces on ARINC-653 Operating System for Safety-Critical Applications

December 10, 2002 - BAE SYSTEMS Controls, a leader in ARINC-653-compliant technology for real-time operating systems, and DDC-I, a provider of tools and services for embedded systems, will jointly develop a platform for writing safety-critical software. The agreement calls for the creation of SCORE-653, combining BAE SYSTEMS CsLEOS Real-time Operating System (RTOS) with the DDC-I SCORE® (Safety-Critical Object-oriented Real-time Embedded) multi-language development environment.

The integration combines a development environment and an ARINC-653-compliant RTOS, both designed for use in high-integrity embedded systems. SCOREs multi-language support will allow the Ada and C programming languages to be used in application development. The combination of Ada's reliability, provided through strong typing

and rigorous checks, and the builtin safety features of the CsLEOS RTOS provide an ideal platform for development of software for safety-critical uses.

The new environment generates PowerPC cross applications and is supported on Solaris and Windows NT. It also will encompass multi-language compilers, a graphical user interface, and multi-language debugger support.

"With more than 50 years of aircraft flight control experience behind it, our CsLEOS operating system is an ideal commercial, off-the-shelf solution to many safety-critical systems requirements," said James Scanlon, president of BAE SYSTEMS Controls. "We are pleased to team with DDC-I, a company respected for its work in embedded systems, to offer customers easier access to this important technology."

DDC-I is seeing increasing demands for software solutions that offer reliable time and space partitioning. "The ARINC-653 standard defines operating systems that meet these requirements," said Dr. Ole Oest, president and CEO of DDC-I Inc. "SCORE-653 will give our customers a well-integrated tool set to enable the development of highly reliable, DO-178B Level A-certifiable applications."

Designed from the beginning to implement ARINC-653 brick-wall partitioning and to be certifiable to the highest FAA DO-178B safety level, the CsLEOS RTOS ensures that safety-critical functions are protected from other processes running on the same hardware. This structure also makes it possible to add, revise, and test system functions without recertifying the entire application.

The first development environment to offer multi-language, multi-target, and multi-host capabilities based on non-proprietary open-system standards, SCORE® addresses the increasing need among project developers to create and combine reusable software components often written in different languages such as Ada and C targeting different microprocessors and created on different development platforms. [...]

BAE SYSTEMS Controls, an operating unit of BAE SYSTEMS North America, is a leading supplier and integrator of electronic flight and engine control systems for defense and commercial aircraft and launch vehicle applications. It also provides integrated avionics for aircraft applications and complete unmanned aerial vehicle systems and is a leader in electronic controls for the locomotive industry and the emerging heavy-duty hybrid-electric vehicle market. Controls operates facilities in Johnson City, New York; Santa Monica and Ontario, California; Fort Wayne, Indiana; and Redmond, Washington. [...]

DDC-I is a global supplier of software development tools, custom software development services, and legacy software system modernization. DDC-I's customer base is an impressive "whos who" list in the commercial, military, aerospace, and safety-critical industries. Tools include compiler systems and run-time systems for Ada, JOVIAL, C, and C++ applications development. [...]

For further information, please contact: Larry Stone, BAE SYSTEMS Controls, Tel: 607-770-3944, Mobile: 607-761-9553, lawrence.w.stone@baesystems.com; Jennifer Sanchez, DDC-I Inc., Tel: 602-275-7172 ext. 201, jc@ddci.com. [...]

DDC-I - SCORE IDE Supports Windows Host Environments

URL:

http://www.ddci.com/news_current_issue.shtml

Subject: DDC-I Online News, January 2003, Vol. 4 Issue 1

DDC-I's SCORE® Integrated Development Environment Now Offers Full Support for Windows Host Environments

Phoenix, Arizona and Lyngby, Denmark, January 15, 2003 -- DDC-I announces the addition of native support for the popular Windows computing platforms, keeping pace with the ongoing migration of safety-critical real-time embedded system developers to the latest network and enterprise computing platforms.

"SCORE's primary purpose is to provide an easy-to-use environment that guarantees maximum software portability and code reusability to safety-critical embedded systems developers," explains David Mosley, DDC-I's SCORE Product Champion. "Support for the latest Windows versions is in direct response to the needs of our customers, who continue to migrate safety-critical software development for a variety of projects to these increasingly popular platforms."

The first IDE to offer multi-language, multi-target and multi-host capabilities based on non-proprietary open system standards, SCORE meets the increasing need among project developers to combine reusable software components, written in different languages, targeting different microprocessors and often created on different development platforms. Support for a native environment compliments SCORE's existing 80x86 and PowerPC cross targets already available on Windows.

Providing a Graphical User Interface (GUI) while retaining a command-line interface option, SCORE incorporates project management tools, online help functions, tool activation and a number of

other efficient features, as well as a universal interface for compilers and tools. The use of open standards means third-party products are easily integrated.

"Cost reduction is the driving force in embedded systems, and SCORE makes the development of embedded applications as efficient and inexpensive as possible. Its greatest strengths are the generation of coherent, modular, and reusable software components, error detection in the early stages of development, and quick and efficient retargeting of existing code," concludes Mosley.

DDC-I - Exclusive Provider for TADS Ada Development Systems

URL:

http://www.ddci.com/news_current_issue.shtml

Subject: DDC-I Online News, January 2003, Vol. 4 Issue 1

DDC-I Exclusive Provider for "TADS Ada Development Systems" Targeting 1750A, 680x0, and i960

Phoenix, AZ and Lyngby, Denmark, January 24, 2003 -- An industry leader in development tools for safety-critical real-time embedded systems, DDC-I announces that effective February 13, 2003, DDC-I will become the exclusive owner of the respected TADS product line targeting the MIL-STD-1750A, Intel i960 and Motorola 68000 microprocessors. TADS, which supports the development of embedded Ada applications, will continue to be available directly from DDC-I and from our distributors worldwide for these targets.

TADS, previously known as the "Tartan Ada Development System", was originally created by Tartan Inc. and subsequently acquired by Texas Instruments. For the past five years DDC-I has distributed and maintained these products for Texas Instruments.

"TADS is now a full-fledged member of the DDC-I product line-up," explains Joyce Tokar, Vice President of Technology at DDC-I and a past principal member of Tartan's technical staff. "Every customer using these proven tools will continue to receive personalized service and support from the company whose central focus is embedded development."

The TADS product line combines highly optimizing compilers with selective linking and modular run-times to generate the most compact code available. In addition to classical optimizations and performance benefits specific to each chip architecture, Ada 83 specific compiler optimizations include data packing, constraint and overflow check elimination and static aggregate initialization.

Green Hills - INTEGRITY RTOS for Intel x86/Pentium

URL:

<http://www.ghs.com/news/221119i.html>

Green Hills Software Boosts Reliability of Intel® x86/Pentium-based Embedded Products

Royalty-free RTOS Provides Application-Critical Protection for End User Products Embedded Systems Conference, Boston, MA: Booth #901, CA. November 19, 2002 -- Green Hills Software, Inc. today announced the availability of its royalty-free INTEGRITY® 4.0 real-time operating system (RTOS) for embedded products based on the Intel x86/Pentium architecture. Featuring a memory-protected micro-kernel architecture that guards against viruses, hackers, and errant code, INTEGRITY provides applications with guaranteed resource availability for both CPU time and memory, yet still delivers hard real-time response, making it ideal for demanding embedded applications. INTEGRITY gives designers using Intel X86/Pentium processors a fast, secure, ultra-reliable, royalty-free target environment for medical, process control, telecommunications, consumer electronics, and mil/aero products, resulting in reduced development time and lower per-unit cost.

INTEGRITY is a small, fast, deterministic RTOS designed for applications that require high reliability, small size, and real-time responsiveness. Utilizing the hardware memory protection facilities of the x86/Pentium MMU, INTEGRITY builds a firewall between the kernel and user tasks that prevents errant or malicious tasks from corrupting user data, the kernel, interprocess communications, device drivers and other user tasks. INTEGRITY also supports distributed processor designs and provides hooks that make it possible to debug and upgrade running systems in the field.

"Until now, embedded system developers using Pentium-class processors had little choice when it came to selecting a robust RTOS environment for their designs," said John Carbone, vice president of marketing for Green Hills Software. "But with the availability of INTEGRITY's total solution for embedded X86/Pentium products, system architects and application programmers now can take advantage of the industry's most complete royalty-free RTOS, compiler and debugger solution, allowing development teams to create the best electronic products in the least time and at the lowest cost."

Frank Willis, vice president of business development for SBS Technologies comments, "With INTEGRITY, our customers now can develop and deploy

much more secure and reliable systems for the most demanding telecommunications, medical, and aerospace/defense applications. INTEGRITY's development tools also offer unparalleled in productivity, enabling our customers to get to market faster with a lower development cost."

INTEGRITY provides comprehensive I/O and file system support, including a USB stack, NFS client, UNIX-compatible, hierarchical file system, flash file system, POSIX API, and drivers for I2C, CompactPCI and VMEbus backplanes, clocks, timers, and standard PC peripherals. INTEGRITY also provides comprehensive networking support, including such protocols as TCP/IP, TCP/IP-lite, UDP, RLOGIN, BSD sockets, PPP, FTP, TFTP, DNS, DHCP, SNMP, and web server. INTEGRITY also supports a full line of embedded Internet software products and applications that make it easy to add web connectivity and management to target systems, and a small, fast, royalty-free embedded graphics solution.

INTEGRITY's graphical ResourceAnalyzer(TM) aids product development and debugging by greatly enhancing visibility into applications and the kernel as they execute on the target system. ResourceAnalyzer displays CPU execution at the task and address space level, stack usage for tasks, and memory use for address spaces. INTEGRITY's real-time EventAnalyzer(TM) further enhances real-time visibility by enabling programmers to log and monitor system and application events in real time without disrupting program execution. Operating like a high-level logic analyzer, EventAnalyzer displays all INTEGRITY context switches, API calls, and interrupts in a time-relative manner using intuitive ICONs. EventAnalyzer is also tightly integrated with Rational's Rose® RealTime, enabling programmers to view Rose RealTime state transitions and INTEGRITY events.

INTEGRITY is seamlessly integrated with Green Hills Software's MULTI® and AdaMULTI® IDE. Together with the Green Hills family of optimizing Ada95/C/C++/EC++ compilers, MULTI automates all aspects of embedded software development for x86/Pentium processors, including editing, source-level debugging, program building, run-time error checking, version control, and code/performance optimization. MULTI also features ISIM(TM), an INTEGRITY simulator that enables programmers to develop and test their x86 code on a PC or workstation without the need for target hardware.

INTEGRITY is available now for the Intel x86/Pentium architecture and works out of the box with PC-compatible ISA/EISA, PCI, and PC/104 boards. In

addition, Green Hills offers a porting kit and turnkey porting services for designers who want to port INTEGRITY to other platforms.

About Green Hills Software, Inc.

Founded in 1982, Green Hills Software Inc. is the technology leader for real-time operating systems and software development tools for 32- and 64-bit embedded systems. Green Hills Software's royalty-free INTEGRITY® real-time operating system and the ThreadX real-time operating system, fully integrated with its market leading compilers and MULTI® Integrated Development Environment, provide a total development and run-time solution that addresses both deeply embedded and maximum reliability applications.

Green Hills Software is headquartered in Santa Barbara, CA, with European headquarters in the United Kingdom. For more information on Green Hills Software products, call 805-965-6044, email sales@ghs.com.

[...]

For More Information Contact: Green Hills Software, Lynn J. Robinson, (805) 965-6044, lynnr@ghs.com; Davis-Marrin Communications, Ken Marrin, (781) 581-8370, kmarrin@davismarrin.com

I-Logix - Rhapsody 4.1 Delivers Rhapsody In Ada

URL:

http://www.ilogix.com/news/press_detail.cfm?pressrelease=2002_11_18_104446_447387pr.cfm

I-Logix' Rhapsody 4.1 Delivers Rhapsody In Ada, The First UML Rule-Based Visual Application Development Platform

Rhapsody 4.1 Platform Breaks New Ground In Usability And significantly Enhances Analysis Capabilities for Model-Based Development

November 18, 2002 - Boston, Mass. - I-Logix Inc., the premier provider of enterprise solutions for embedded applications development, today introduced Rhapsody® 4.1, the newest version of its award winning Unified Modeling Language (UML) based application development platform. Rhapsody 4.1 delivers the Rhapsody in Ada Developer Edition, for Ada 83 and Ada 95, the first visual application development platform based on the UML for the Ada systems and software developer. In addition, 4.1 incorporates significant analysis capabilities and usability enhancements aimed at improving systems design capability and developer productivity.

Rhapsody in Ada is the only system and software design environment based on UML that offers complete coverage of the

systems and software development process including requirements capture and analysis, systems design, full production code generation, design-level debugging and test for Ada applications. Optimized for the development of real-time embedded applications in the military, aerospace and transportation industries, Rhapsody in Ada incorporates a rule-based reverse engineering capability that provides the flexibility and control to enable Ada developers to leverage legacy code in new or evolving applications. The Rhapsody in Ada rule-based code generation has been architected to enable end-users a high level of customization for the generated Ada code, thus allowing them to conform to strict, well-defined development standards. Furthermore, requirements based testing ensures that a design meets requirements by using the models developed throughout the development process as test systems. By shifting the focus of work from coding and debugging to design, increasing the opportunity for design re-use and simplifying communication among design team members, Rhapsody in Ada boosts development team productivity and dramatically improves time-to-market.

Usability and feature enhancements focused on improving systems engineering capabilities and increasing developer productivity are a major part of the Rhapsody 4.1 release. Users are now able to locate properties more quickly using the tree-view browser and new property filters. The sequence diagram editor has been re-architected to facilitate both simple and complex scenario development, with an enhanced drawing capability. Activity Diagrams have been augmented to support modeling of multiple concurrent processes via swimlanes. Hyperlinks have been added so users can link to external documents or web pages from any location within the Rhapsody design. To further facilitate both small- and large-scale collaboration, differencing and merging has been enhanced for graphical comparison and integration, enhancing the ability for multiple developers to work on a single design.

"The delivery of Rhapsody in Ada, developed in collaboration with major aerospace companies, provides a major advantage to developers of military, aerospace and transportation applications. They are now able to develop safety-critical applications, customizing code as needed, with confidence that their design is always in synch with requirements," said Neeraj Chandra, Senior VP of Marketing & Corporate Development. "In addition, Rhapsody 4.1 has major usability and systems design enhancements, which deliver on our commitment to improving user productivity at every level. It is critical

that our customers are able to work as efficiently and as easily as possible."

About I-Logix Inc.

Founded in 1987, I-Logix is a venture backed software company that provides enterprise solutions for real-time embedded applications development in the growing pervasive computing market. I-Logix solutions significantly compress systems and software development cycles while improving product quality. These products allow engineers to graphically model the behavior and functionality of their embedded systems, analyze and validate the system and automatically generate production quality code in a variety of languages. I-Logix uniquely integrates and associates the entire design flow from concept to code across an enterprise using both conventional and collaborative Web-enabled technology.

I-Logix is a member of the Object Management Group (OMG), the Bluetooth SIG, the International Council of Systems Engineers (INCOSE), a founding member of the Embedded Linux Consortium and a co-author of the Unified Modeling Language (UML). [...] The company is headquartered in Andover, Mass., and has sales offices and distributors throughout the USA, Europe and the Far East. I-Logix can be found on the Internet at www.ilogix.com.

For more information on any of the company's embedded software products, please feel free to contact us.

Contacts: Carrie Kirby, Manager, Marketing Communications, I-Logix, 978-682-2100, carrie@ilogix.com

OAR - GNAT/RTEMS Cross-Compiler System

From: joel@OARcorp.com (Joel Sherrill)

Date: 6 Jan 2003 14:24:36 -0800

Subject: Re: Porting the GNAT runtime to an embedded target

Newsgroups: comp.lang.ada

[On questions about porting the GNAT runtime to embedded platforms. -- dc]

> Take a look at RTEMS (www.oarcorp.com). If you can do your development on Linux there are RPMS available for GNAT/RTEMS. [...]

The last batch of gcc 3.2.1 RPMS included GNAT/RTEMS for the i386, sparc, powerpc, and mips targets.

> It may be possible to make GNAT/RTEMS work under cygwin. [...]

I haven't tried it but would think that if you could build the native gnat from gcc 3.2.1, then you should be able to build GNAT/RTEMS. You need to use the same version native compiler as you are building cross.

[See also "OAR - RTEMS Operating System" in AUJ 22-4 (Dec 2001), pp.209-210. -- dc]

Praxis Critical Systems - SPARK Incorporates Ravenscar Features

From: Vinzent Hoefler

<ada.rocks@jlfencey.com>

Date: Tue, 04 Feb 2003 10:57:38 -0500

Organization: JeLlyFish software

Subject: Re: SPARK or Ravenscar

Newsgroups: comp.lang.ada

> SPARK typically involves more dedication to proofs but Ravenscar is not intended to be low integrity I would think. Do you think that SPARK is typically safer (for space work)?

Well, probably you cannot really compare the two.

Ravenscar is just a profile on what Ada features to use and what not to use. In this sense the concept (subset of existing language) might be seen similar to SPARK. But SPARK is a Ada-subset language with additional annotations for automatic proof and in some terms it is far more restricted than plain Ravenscar.

So, from what I know, I'd say most if not all SPARK programs would conform to Ravenscar, but a Ravenscar conforming program is not necessarily written in SPARK.

So, in this view SPARK programs look safer, yes. :)

From: Phil Thornley

<phil.thornley@baesystems.com>

Date: Wed, 5 Feb 2003 08:37:38 -0000

Subject: Re: SPARK or Ravenscar

Newsgroups: comp.lang.ada

SPARK and Ravenscar can't be compared because they address different aspects of the language: the Ravenscar profile is explicitly silent on the sequential part of the language; the SPARK language is (currently) limited to sequential programs.

So there is an excellent case for using them both - restrict the tasking features used to conform to Ravenscar and analyse each task as a separate SPARK program.

In fact this is such a good idea that the SPARK developers are planning to add Ravenscar compliant features to the next version of SPARK. :-)

The guide to Ravenscar has recently been completed by the HRG. It is published initially as a University of York report: <ftp://ftp.cs.york.ac.uk/reports> and click on YCS-2003-348.pdf and will be offered to WG9 as a potential ISO technical report.

Phil Thornley, BAE SYSTEMS

From: rod.chapman@praxis-cs.co.uk (Rod Chapman)

Date: 5 Feb 2003 04:01:49 -0800

Subject: Re: SPARK or Ravenscar

Newsgroups: comp.lang.ada

I'm pleased to say that the next release of the SPARK language and toolkit will incorporate the Ravenscar Profile as a core part of the SPARK language. Simply put, SPARK will be a strict subset of the Ravenscar Profile incorporating library level tasks, protected objects, suspension objects, Ada.Real_Time and so on.

The language will, of course, be amenable to all the existing forms of static analysis implemented by the Examiner, but will add:

1) Analysis to show that programs are free from Ravenscar-defined erroneous behaviour, exceptions, bounded errors and so on. For example, SPARK will be statically free from priority ceiling violations, entry queue length violations, and so on.

2) Information flow analysis at the partition-level.

3) and much more...

We will be presenting a tutorial on the combined "RavenSPARK" language and its analysis at Ada Europe in Toulouse in June.

Rod Chapman, SPARK Team, Praxis Critical Systems

Rational - IBM Acquires Rational

From: Richard Riehle

<richard@adaworks.com>

Date: Fri, 06 Dec 2002 18:47:56 -0800

Organization: AdaWorks Software Engineering

Subject: IBM Acquires Rational Ada

Newsgroups: comp.lang.ada

Just announced today was the 2.1 billion dollar purchase of Rational by IBM. One can only wonder what will happen to the Ada compiler products. [...]

From: Richard Riehle

<richard@adaworks.com>

Date: Sat, 07 Dec 2002 17:46:43 -0800

Organization: AdaWorks Software Engineering

Subject: Re: IBM Acquires Rational Ada

Newsgroups: comp.lang.ada

> Memo is here:

http://www.internalmemos.com/memos/memo_details.php?memo_id=1145

Alas, no hint of what will become of Rational Ada. If anyone at IBM realizes the power of the Rational Ada product, it could be great for Ada and for IBM. [...]

From: Steven Deller

<a101.deller@smsail.com>

Date: Sun, 8 Dec 2002 08:45:08 -0600

Organization: Smooth Sailing LLC

Subject: RE: IBM Acquires Rational Ada

Newsgroups: comp.lang.ada

I'd bet that when the smoke clears, Rational Ada will live on pretty much as now. It is possible that IBM will actually add support for it.

It can't go away -- too many large users of Rational Ada (who are also IBM customers) will bring pressure to keep support of the product.

Granted that's a guess, but it is an "educated" guess. If nothing else, it should make you a bit more optimistic. [...]

Rational - Future Commitments

From: Glenn, Eddie <cav@Rational.Com>

Date: Fri, 20 Dec 2002 11:43:39 -0800

Subject: FW: A Message from Mike Devlin, CEO of Rational Software

To: apex_forum@Rational.Com, apex-announcements@Rational.Com

(This message is being sent on behalf of John Lambert, VP & General Manager of the product group responsible for Apex)

Below, please find a message from Rational's CEO, Mike Devlin, addressing some initial questions you may have had regarding the announced intent of IBM to acquire Rational Software. This same message can be found on the Rational & IBM websites:

Rational: <http://www.rational.com/news/dotnet-customers.jsp>

IBM:

<http://www3.ibm.com/software/swnews/swnews.nsf/n/mmaa5gx3fr#15>

As you will read, Rational and IBM remain very strongly committed to our technical and embedded products - including our Apex family - and together are intent upon further enhancing our support for customers in these sectors.

The team here is very excited about all the fresh opportunities this new phase in Rational's business evolution will bring, and we'll continue to keep you in the loop as things move forwards.

From: Rational News

[mailto:RationalNews@rational.com]

Sent: Wednesday, December 18, 2002 2:22 PM

Subject: A Message from Mike Devlin, CEO of Rational Software

To: RationalAnnouncementService@listserv.ce.rational.com

A Message from Mike Devlin, CEO of Rational Software

Over the past several days, many customers have approached Rational with questions about the impact of the pending IBM acquisition of Rational Software. I wanted to take this opportunity to give you the same guidance and assurances we have provided them.

For the past twenty-one years, Rational has built its business by serving a variety

of customers building many different types of software on a mix of platforms and languages. We intend to continue on this path. In addition, in a recent FAQ on the Rational acquisition, IBM clearly states its position on the matter. The FAQ is available at <http://www-3.ibm.com/software/swnews/swnews.nsf/n/mmaa5gx3fr>. [...]

Rational is committed to the ongoing development and evolution of our technical and embedded software development solutions on the broadest range of embedded development and execution platforms.

The IBM FAQ recognizes that "Rational is a leader and has a deep heritage with customers building software for technical software products and systems" and asserts that "We intend to continue to support those customers and invest in our solution to help them build better software faster." [...]

Rational - Apex Embedded 4.2.1 for LynxOS Sun Solaris to PowerPC Family

From: Glenn, Eddie <cav@Rational.Com>
Date: Mon, 25 Nov 2002 15:16:54 -0800
Subject: Rational Apex Embedded for LynxOS Sun Solaris to PowerPC Family, version 4.2.1 is available by FTP
To: apex-announcements@rational.com

[In all Rational URLs below, substitute <rel> by <cross>/releases/sol,

<doc> by <cross>/documents/unix/Rexec, <cross> by <ftp>/apex_cross/, and <ftp> by <ftp://ftp.rational.com/public/--dc>]

Product: Rational Apex Embedded for LynxOS

Version: 4.2.1

Platform: Sun Solaris to PowerPC

URL: <rel>/power/lx/4.2.1

Rational Apex Embedded for LynxOS, version 4.2.1, is an update release that supports targets running LynxOS 4.0. It should only be installed by customers wishing to use LynxOS 4.0.0 with Apex Embedded. [...]

Follow this link for Rational Apex Embedded for LynxOS download and installation instructions. [...]

<ftp>/standard.msgs/install_instructions.html

Rational - Apex Embedded 4.2.0 for Rational Exec Sun Solaris to MIPS Family

From: Glenn, Eddie <cav@Rational.Com>
Date: Fri, 24 Jan 2003 10:12:49 -0800
Subject: Rational Apex Embedded for Rational Exec Sun Solaris to MIPS

Family, version 4.2 is now available via FTP

To: apex-announcements@rational.com

Product: Rational Apex Embedded for Rational Exec

Version: 4.2.0

Platform: Sun Solaris to MIPS

URL: <rel>/mips/rt/4.2.0

This release is pending Generally Available (GA) status as it goes through the final steps of the manufacturing process. We anticipate that this will be complete within the next 30 days. Once this release reaches GA status, it will be available for shipping. Until then, it is being provided on this FTP server for immediate access.

Follow this link for Rational Apex for Rational Exec download and installation instructions. [...]

<ftp>/standard.msgs/install_instructions.html

Rational Apex Exec 4.2.0 Release Note for MIPS:

<doc>/relnote.4.2.dir/rexec_relnote_mips/rexec_release_noteTOC.html

<doc>/relnote.rexec.mips.4.2.ps.Z

Rational - Apex 4.2.0b Patches for Solaris 64 bit support

From: Glenn, Eddie <cav@Rational.Com>
Date: Mon, 27 Jan 2003 08:02:10 -0800
Subject: Patches for Apex 4.2.0b - Solaris 64 bit support
To: apex-announcements@rational.com

A set of patches has been released to provide support for 64 bit integers and addresses on Solaris. To use these patches you must be running a 64-bit version of Solaris on a hardware platform that supports 64-bit integers and addressing (ie: SPARC v9).

With Solaris an application is either 32 bit integers and addresses, or it is 64 bit integers and addresses. There is no I64/A32 mode.

In general on any platform where there is a choice, 64 bit applications will run slower than the equivalent 32 bit application, simply because more data is being moved around. 64 bit applications will also use more memory as all addresses (ie: pointers and access types) are now 64 bits long.

The patch set consist of 3 patches

200301062-1 - 64 bit Ada Compiler and Runtimes

200301063-1 - 64 bit Ada predefines

200301064-1 - 64 bit C++ Compiler and Models

[...] The patches are located at:

<ftp>/apex/releases/sol/patches/Ada/200301062-1 (44MB)

<ftp>/apex/releases/sol/patches/Ada/200301063-1 (137MB)

<ftp>/apex/releases/sol/patches/C++/200301064-1 (22MB)

If you need support for Solaris 64 bit and cannot download these large patches, please contact ask_apex@rational.com to arrange for special media to be built and shipped.

RR Software - Janus/Ada for Windows

From: Randy Brukardt
<randy@rrsoftware.com>

Date: Tue, 7 Jan 2003 12:56:11 -0600

Subject: Re: advantages or disadvantages of Ada over Pascal or Modula

Newsgroups: comp.lang.ada

> This sounds good. But - how expensive is the compiler? Would it be a choice for a hobbyist, or is it far beyond? [...]

The Janus/Ada Personal Edition starts at \$195 for Windows. The Combo (which includes Claw) is \$295. Probably we should charge more, but our root are as hobbyists ourselves (in the days before Open Source).

For all of the prices on RR Software's Ada products, look at:

<http://www.rrsoftware.com/html/companyinf/prices.htm>

Semantic Designs - DMS Software Reengineering Toolkit

From: Ira Baxter
<idbaxter@semdesigns.com>

Date: Tue, 5 Nov 2002 19:30:37 -0600

Subject: Re: Building documentation from comments

Newsgroups: comp.lang.ada

> [...] Any other programs for extracting comments from Ada sources?

The DMS Software Reengineering Toolkit is generalized compiler technology for parsing / analyzing / transforming / prettyprinting many source languages, including Ada. See <http://www.semdesigns.com/Products/DMS/DMSToolkit.html>.

It could be very easily configured to extract arbitrary information from Ada comments. Its Ada83/95 parser builds a full syntax tree decorated with the comment text. Walking over the tree can visit all the comments, what you do with them after than can be arbitrary.

A "trivial" application is to print the source code, with the comments, nicely formatted. See

<http://www.semdesigns.com/Products/Formatters/AdaFormatterExample.html>.

[And upon a question about ordering and pricing: -- dc]

In general,

<http://www.semdesigns.com/Products/index.html>, see top of page. The DMS Toolkit is quoted on request, as it comes with lots of options.

Ira D. Baxter, Semantic Designs, Inc.

SofCheck - New Company in Ada Business

*From: Robert A Duff
<bobduff@shell01.TheWorld.com>
Date: Sat, 23 Nov 2002 18:02:48 GMT
Subject: Re: GNAT Programming System availability?
Newsgroups: comp.lang.ada*

> How can one get more information about SofCheck? What does SofCheck do? Are they in the Ada business?

SofCheck, Inc. is a new company owned by Tucker Taft, Mireille Gart, Sheri Bernstein, and me, all former employees of AverCom (which was previously AverStar, which was previously Intermetrics).

Yes, we are in the Ada business. We sell and support the AdaMagic compiler technology, formerly owned by AverCom. This includes an Ada-to-C translator, which can be used as an Ada compiler on any target that has a C compiler. AdaMagic also supports compilers targeted directly to machine code for various machines.

SofCheck is also in the business of static analysis tools for various languages, including Ada.

For more information, contact Tucker Taft, President of SofCheck (stt@sofcheck.com).

*From: Robert A Duff
<bobduff@shell01.TheWorld.com>
Date: Tue, 21 Jan 2003 19:16:22 GMT
Subject: Re: advantages or disadvantages of Ada over Pascal or Modula
Newsgroups: comp.lang.ada*

[About the Ada-to-C translator: -- dc]

> What do you test, debug, document and so on: Ada or C-code?

You test the machine code, of course.

You debug either at the Ada level or the C level -- your choice. The Ada compiler can insert #line directives and whatnot into the generated C code, so that one can set breakpoints on Ada lines and so forth. It's not perfect, but it works. In some cases, you need to understand the mapping from Ada names to C names, for example (which is fairly obvious). On some targets, the debugger has been trained to understand Ada better.

You document the Ada code. The C code is just an intermediate language used by the compiler, and can be ignored by the programmer, except in some debugging situations.

SofCheck also supports other targets for Ada that do not use C as an intermediate language.

Robert A. Duff, Vice President of Engineering, SofCheck, Inc.

PTLogica - TwinText Source Code Documentation Tool

*From: pabtro <pabtro@ptlogica.com>
Date: Mon, 06 Jan 2003 19:48:17 GMT
Subject: ANN: TwinText 1.3 Released - Source Code Documentation Tool
Newsgroups: comp.lang.ada*

TwinText 1.3 Released

TwinText is an easy to use source code documentation solution. It generates customizable HTML and HTML Help directly from naturally looking code comments. TwinText supports virtually all programming languages.

<http://www.ptlogica.com/TwinText/overview.htm>

Ada and Linux

GNAT 3.15p Linux Distribution

*From: file13@qliptho.zzn.com (file13)
Date: 21 Nov 2002 07:51:08 -0800
Subject: Re: GNAT 3.15p on various Linux versions
Newsgroups: comp.lang.ada*

> The name of the GNAT 3.15p tarball for Linux indicates that it was produced under Red Hat Linux 7.1. Has its operability been proven on other versions and flavors of Linux, [...]

[Jean-Pierre Rosen replied:] Works fine on my Mandrake 7.2.

It's just dandy on me Mandrake 8.0 and 8.2 boxes....

*From: Georg Bauhaus <sb463ba@11-hrz.uni-duisburg.de>
Date: Wed, 5 Feb 2003 18:16:18 +0000 (UTC)*

*Subject: Re: Catching exceptions in debugger (Unix)
Newsgroups: comp.lang.ada*

> There is no 3.15p package for Debian.

Installing 3.15p on Debian in a location of your choice is a trivial procedure, if you use the "standard" GNU/Linux tar.gz-Distribution. There is also a Debian package containing GCC 3.2 incl Ada. None of them steps on any other's feet if you follow the installation instructions.

Ada and Microsoft GNAT for DOS Update

*From: Gautier de Montmollin
<gdemont@hotmail.com>
Date: Mon, 04 Nov 2002 16:23:54 +0000
Subject: Re: Ada and Win98
Newsgroups: comp.lang.ada*

> I saw no mention of "Ez2load" in all this discussion. [= stable GNAT for DOS distribution. -- dc] Is that item no longer up-to-date?

Yes and no: it functions (it is a GNAT 3.10p), there are newer versions up to GNAT 3.14p available. But it is a DOS thing, not a Windows one. Of course it runs on Windows (more precisely, on its DOS Virtual Machine), as well as executables.

More is there :

<http://www.mysunrise.ch/users/gdm/gnatdos.htm>

CPU Usage for Windows NT

*From: Sim Con <sicon@hotmail.com>
Date: Thu, 7 Nov 2002 15:44:08 +0000 (UTC)*

*Subject: My little contribute: CPU Usage for Win NT
Newsgroups: comp.lang.ada*

Hello! My little contribute to everyone who need/like a CPU usage package for Windows NT. Hope it helps like did for me. Any comments would be useful, thanx!

<http://www.geocities.com/Nighterya/CpuMeterNTV2.0.zip>

A# - Port of GNAT to .NET

*From: Martin Carlisle
<carlisle@web2news.net>
Date: Tue, 26 Nov 2002 22:03:37 +0100
Subject: New Ada compiler for .NET
Newsgroups: comp.lang.ada*

We have completed a port of the GNAT compiler to the .NET framework, which is based on the ACT tool JGNAT. More information and downloads available from

http://www.usafa.af.mil/dfcs/bios/mcc_html/a_sharp.html

This tool will be presented at SIGAda 2002 in Houston, TX, USA, on 11 December 2002.

[See also "Ada for .NET Effort in Progress" in AUJ 23-4 (Dec 2002), p.206. -- dc]

Martin C. Carlisle, PhD, Associate Professor and Advisor-in-Charge, Department of Computer Science, United States Air Force Academy

URL:

http://www.usafa.af.mil/dfcs/bios/mcc_html/a_sharp.html

Subject: Ada for .NET

Welcome to the A# home page! [...]

A# is a port of Ada to the Microsoft .NET Platform. A# is freely distributed by the Department of Computer Science at the United States Air Force Academy as a service to the Ada community under the terms of the GNU general public license.

A# has been fully integrated into AdaGIDE, a leading open-source IDE for Ada under Windows. A .NET port of the multi-platform open-source GUI Design tool RAPID has been completed. Negotiations are in progress with Microsoft to include Ada in Visual Studio .NET.

Read more about the A# project in our SIGAda 2002 paper.

[http://www.usafa.af.mil/dfcs/bios/mcc_html/asha_rp.pdf -- dc]

Features

* Extract Ada specification files directly from .NET DLLs (MSIL2Ada)

* Compile Ada programs into .NET executables (MGNAT)

Download & Rebuilding from sources:

[see online version of this web-page -- dc]

Authors: Dr. Martin C. Carlisle, Lt Col Ricky Sward, Capt Jeff Humphries.

Comments, suggestions, and bug reports are welcome. If you have a comment, suggestion or bug report, send email to: Martin.Carlisle@usafa.af.mil.

From: David C. Hoos
<david.c.hoos.sr@ada95.com>

Date: Wed, 29 Jan 2003 11:56:42 -0600
Subject: Re: GNAT and cross-platform GUI
Newsgroups: comp.lang.ada

[...] there is also a European "mirror" for the downloads at <ftp://sunsite.informatik.rwth-aachen.de/pub/mirror/ftp.usafa.af.mil/pub/dfcs/carlisle/asharp/> [...]

From: Carlisle Martin C Dr USAFA/DFCS
<Martin.Carlisle@usafa.af.mil>

Date: Thu, 5 Dec 2002 16:10:46 -0700
Subject: Re: Future of Ada
To: team-ada@acm.org

> [...] complete lack of support or mention of Ada by major vendors like Microsoft. [...]

Ada isn't totally off of Microsoft's scope. See <http://www.gotdotnet.com/team/lang/> and you'll find Ada as the first language link (OK, so the alphabet helped out a little there :-).

I am in the midst of currently successful negotiations to add Ada to MS Visual Studio .NET. Since this email is probably shameless self-aggrandization anyway, I'll point out that I'll be presenting Ada for .NET at SIGAda 2003 in Houston next week.

AdaGIDE 7.00 - Ada GUI IDE for Windows

From: gdemont@hotmail.com (Gautier)
Date: 3 Dec 2002 12:34:14 -0800
Subject: AdaGIDE 7.00 release (Ada GUI IDE for Windows 95,98,ME,NT,2000,XP)
Newsgroups: comp.lang.ada

AdaGIDE 7.00 release (Ada GUI IDE for Windows 95,98,ME,NT,2000,XP)

AdaGIDE (the Ada GNAT Integrated Development Environment) is an interface to the GNAT compiler featuring a color context-sensitive editor and code reformatter. It runs on Windows 95,98,... and Windows NT,... series.

URL:
http://www.usafa.af.mil/dfcs/bios/mcc_html/adagide.html

Main improvements in AdaGIDE 7.00 compared to version 6.52: support for .NET target (See A# project (3)); goto declaration now switches from spec to body, too; suggestions added when a '.' or '(' is pressed (Ada. -> ...); remembers last size and location of window; longer recent file list.

[See also "AdaGIDE 6.52 - Ada GUI IDE for Windows" in AUJ 23-2 (Jun 2002), p.76. -- dc]

Gautier (1) - on behalf of Martin (2)

(1)
<http://www.mysunrise.ch/users/gdm/gsoft.htm>

(2)
http://www.usafa.af.mil/dfcs/Faculty/bio/dfcs_bio_carlisle.htm

(3)
http://www.usafa.af.mil/dfcs/bios/mcc_html/a_sharp.html

Ada Starter CD-ROM for MS-Windows

From: Roger Gariépy
<rgariépy@rocler.qc.ca>
Date: Sun, 22 Dec 2002 15:37:19 -0500
Subject: Re: Ada starter cd-rom for MS-Windows?
To: team-ada@acm.org

[See also "AIDE - Ada Instant Development Environment for Windows" in AUJ 23-3 (Sep 2002), p.148. -- dc]

> I consider to try to get my cousin hooked on programming in Ada. [...] find the right tools for Ada development on a MS-Windows system. I have decided to give him the book "Object-Oriented Software in Ada 95", but in addition to that I would like to put together a cd-rom with good, free and useful tools and libraries. [List of tools deleted -- dc]

All of what you are asking exist.

It was done by Stéphane Rochebrune, it's called: AIDE (Ada Instant Development

Environment) in the GNU General Public License universe. I am using it on Windows 2000. You can obtain a CD-Rom [...] or download it (as I have done) and burn your CD.

To have more information got to:
<http://eig.unige.ch/lii/Aide.htm> to FTP:
<ftp://129.194.186.90/aide>

It will be better if you read French [...], but most of the Ada documentation is in English.

If I remember correctly you can work directly from the CD or copy the CD on a disk drive and you don't need to configure anything. It's ready to work immediately if you put it in C:\Aide and copy the DLL in the OS directory.

The editor of choice is AdaGIDE (modified so that it don't need installation and parameters to find GNAT, etc.) but Emacs is there also. [...]

Win32_Shell - Launching a Program on Windows

From: gdemont@hotmail.com (Gautier)
Date: 13 Jan 2003 02:37:11 -0800
Subject: Re: ouverture fichier
Newsgroups: fr.comp.lang.ada

[Translated from French: -- dc]

Can we suppose that it's on Windows? In this case you can use Win32_Shell.ad * win_paqz.zip in my files (cf below). Demo in Win32_Shell_Test.adb. That corresponds to the "start" command.

http://www.mysunrise.ch/users/gdm/gsoft_fr.htm

[Another option is to use the more portable OS_Services package from J-P. Rosen, mentioned earlier in this news section. -- dc]

References to Publications

DDC-I Online News

[Extracts from the table of contents. See elsewhere in this news section for selected items. -- dc]

From: JC <jcdk@ddci.com>
Date: Tue, 10 Dec 2002 17:07:19 -0700 (MST)
Subject: Real-Time Industry Updates - News from DDC-I
To: N9DK Dec 2002 Online News
<jcdk@ddci.com>

DDC-I Online News, December 2002, Volume 3, Number 11. A monthly news update dedicated to DDC-I customers & registered subscribers.

BAE SYSTEMS & DDC-I Join Forces on ARINC-653 Operating System for Safety-Critical Applications. The integration provides a development environment designed for use in high-integrity embedded systems.

Third Party Update. CsLEOS RTOS from BAE SYSTEMS: A Safe Choice!

Tech Talk: Memory Mapped I/O in 80x86 Real Address Mode

For the complete newsletter, go to http://www.ddci.com/news_vol3num11.shtml

From: JC <jcdk@ddci.com>

Date: Tue, 28 Jan 2003 16:38:51 -0700 (MST)

Subject: Real-Time Industry Updates - News from DDC-I

To: O9DK Jan 2003 Online News <jcdk@ddci.com>

DDC-I Online News, January 2003, Volume 4, Number 1. A monthly news update dedicated to DDC-I customers & registered subscribers.

2003 at DDC-I - A Promising Future. Our mission is clear, and it shows!

DDC-I Exclusive Provider for TADS Targeting 1750A, 680x0, & i960. Effective February 13, 2003, DDC-I officially takes ownership of TADS.

SCORE Now Available for Windows. Keeping pace with the ongoing migration of safety critical real-time embedded systems.

For the complete newsletter, go to http://www.ddci.com/news_current_issue.shtml

On-line GNU Ada Run-Time Book

From: Javier Miranda

<jmiranda@iuma.ulpgc.es>

Date: Sat, 14 Dec 2002 19:26:57 GMT

Subject: ANNOUNCE: Free GNU Ada Run-Time Book (version 1.0)

To: undisclosed-recipients

I am glad to announce version 1.0 of my free book on the GNAT Run-Time. It is already available at: <http://gnat.webhop.info> and <http://www.iuma.ulpgc.es/users/jmiranda>. The first address is an "easy-to-remember" access to the second address, which is my personal web page.

[Title: A Detailed Description of the GNU Ada Run-Time. -- dc]

The major contributions of this version are:

- The document is now integrated with the GNAT 3.15p sources.

- 176 hyper-links to the GNAT sources have been added (previous version had 126 hyper-links; the new version has 302 hyper-links). This simplifies the use of the book to read the GNAT sources.

- 23 bugs reported by Jose Gonzalez Rodriguez (student of the University of Almeria, Spain) have been corrected.

- The English text has been improved by professor David Shea (Faculty of Translation and Interpretation, University of Las Palmas de Gran Canaria).

- A new index has been added at the end of the book. This index provides a fast access to any concept discussed in the book.

In addition, the access to the HTML version of the GNAT sources is now faster than in previous beta versions.

Javier Miranda, University of Las Palmas de Gran Canaria, Canary Islands, Spain

[See also same topic in AUJ 23-4 (Dec 2002), pp.206-207, for a full description of the book. -- dc]

From: Javier Miranda

<javiermirandag@netscape.net>

Date: Mon, 16 Dec 2002 11:56:05 +0000

Subject: Re: ANNOUNCE: Free GNU Ada Run-Time Book (version 1.0)

Newsgroups: comp.lang.ada

> Why is the GNAT Reference Manual and User Guide not enough?

The GNAT RM gives a complete description of GNAT specific features which can be used by programmers to improve the efficiency of their Ada programs. It is thus written for users of the compiler.

On the contrary, my book gives a description of the current implementation of the Ada Run-Time (how it uses the POSIX services to implement the Ada semantics). This is a complete different point of view which is important for teaching on systems programming (specially real-time concepts) and compilers.

From: Javier Miranda

<javiermirandag@netscape.net>

Date: Mon, 16 Dec 2002 18:21:59 +0000

Subject: Re: ANNOUNCE: Free GNU Ada Run-Time Book (version 1.0)

Newsgroups: comp.lang.ada

> Thank you very much for this book.

Although I am using it only to alleviate my pathological curiosity of how Ada works internally, the value of your work, not only for this application, is enormous!

The effort invested in writing the book has been also enormous! The only references I had were the GNAT sources and the papers found on the Web written by professor Ted Baker (FSU), some of them now out-of-date. In addition to the technical document, I also had to add the hyper-links to the Run-Time sources. In the beta versions I initially did all the work by hand (a lot of work!). In this version I have written some scripts which facilitate to me this work.

One of the main benefits of the free technology is that we can learn a lot from it (at least I did!). In the case of Ada, GNAT is an excellent compiler which gives many possibilities for research and teaching (not only about Ada but also about POSIX and, in general, Software Engineering). I assume the book has

bugs (and things to improve). From here I request help to have up-to-date documents which give details to a big community of researchers and students which can also learn from it.

Thanks for your comments !

From: javiermirandag@netscape.net

Date: Mon, 16 Dec 2002 16:07:38 -0500

Subject: Re: ANNOUNCE: Free GNU Ada Run-Time Book (version 1.0)

Newsgroups: comp.lang.ada

> I'm just a user [...] more interested in the cases why [my program] does not run like possible bad interinfluences of Ada tasks and POSIX threads...

That's is another thing you can read in the book (chapters 2 and 3). In addition, in chapter 4 you can learn how protected objects are implemented (with POSIX mutexes), and in chapter 6 you can learn how the POSIX signals are handled by the Run-Time to implement the Ada interrupts semantics.

In general, understanding the underlying technology helps you writing better programs.

Embedded Systems Programming Journal - "Intro to Ada" by Ben Brosgol

From: Ann S. Brandon

<abrandon@sover.net>

Date: Mon, 20 Jan 2003 17:11:23 -0500

Subject: [AdaIC] Ben Brosgol publishes "Intro to Ada" in ESPJ

To: announce@adaic.com

A well-known and oft-elected figure in the Ada community, Ben Brosgol, of Ada Core Technologies, has published "Intro to Ada" in the Embedded Systems Programming Journal. It will be the first of a series of articles he writes on Ada for them. Please see the AdaIC Website <http://www.adaic.org> for the link under "News", or "What's New", or "Learn Ada", or the button on the left, "Intro to Ada".

Ann Brandon, Editorial Webmaster
AdaIC, webmaster@adaic.org

URL:

<http://www.embedded.com/2003/0301/>

Subject: Embedded Systems Programming Contents for January 2003

January ESP, Vol. 16 No. 1, January 2003

Table of Contents [...]

Introduction to Ada 95, by Benjamin M. Brosgol

Ada, an internationally standardized object-oriented language, was designed for programming large, long-lived systems, especially those with stringent reliability requirements. This introduction is a survey of Ada's key features, with a

focus on those most useful to embedded programmers. [...]

Online Paper - Add Finalization

From: Grein, Christoph
 <christoph.grein@eurocopter.com>
Date: Mon, 13 Jan 2003 09:21:10 +0100
 (MET)
Subject: Re: Finalization of tagged types - problem
Newsgroups: comp.lang.ada

I've published a few years ago in Ada Letters a paper "Add Finalization" dealing with this problem. You can find it online on my home page

<http://home.T-Online.de/home/Christ-Usch.Grein/Ada/Finalization.html>

[From another message: -- dc]

> So is it your opinion: derive from Controlled if it may be needed in a derived?

No, it's not my opinion, it's a fact that you cannot add controlledness to unlimited types later without getting into an ocean of troubled water. But I would not derive from Controlled just for suspense it might be needed later. When it turns out to be necessary, this is a design change, so you have to go back in any case and rethink the design. [...]

Aviation Week & Space Technology - Updates on Ada Usage

From: Marc A. Criley
 <mcq95@earthlink.net>
Date: Fri, 17 Jan 2003 12:08:31 GMT
Organization: Quadrus Corporation
Subject: B-1B Upgrade with AdaMulti
Newsgroups: comp.lang.ada

> From the 1/13/03 Aviation Week & Space Technology, "AeroBytes", pg 393:

"Boeing is modifying B-1B bombers to carry a greater variety of smart weapons in the Conventional Missions Upgrade Program, which includes replacing the six computers in the avionics, navigation and weapons systems with four more powerful units, each with a pair of PowerPC processor cards. The new computers employ Green Hills Software's Integrity real-time operating system to take advantage of multiple-target capabilities and other new features, and programming will be done in Ada with the company's AdaMulti development environment."

Also, the article immediately preceding this one, about BAE Systems Controls teaming up with DDC-I to develop Score-653 for safety critical embedded systems, mentions that "Ada is considered a reliable language".

[For both topics, see also elsewhere in this AUJ. -- dc]

Yet Another Ada

From: Jean-Pierre Rosen
 <rosen@adalog.fr>
Date: Fri, 17 Jan 2003 15:46:11 +0100
Subject: Encore une nouvelle Ada...
Newsgroups: fr.comp.lang.ada

<http://www.auracan.com/ada-enigma/>

[Ada Enigma is the name of the main figure in this French comic book series. -- dc]

Crosstalk Articles

From: Nielson Mark S Civ OO-ALC/MASEB
 <Mark.Nielson@hill.af.mil>
Date: Sat, 25 Jan 2003 11:23:54 -0700
Subject: The February 2003 Issue of CrossTalk is now available on-line.
To: 'Dirk.Craeynest@offis.be'
 <Dirk.Craeynest@offis.be>

The February 2003 issue of CrossTalk, The Journal of Defense Software Engineering is now available on our Web site at: www.stsc.hill.af.mil.

This month we jump into in the myriad sea of choices presented in our theme "Programming Languages." We begin with a historical primer of programming language evolution, then offer some views on the language selection process.

Our first article is "Evolutionary Trends of Programming Languages" by Lt. Col. Thomas M. Schorsch and Dr. David Cook. The authors discuss the needs and forces that have shaped the evolution of programming languages, then they examine the various evolutionary paths of current languages.

Our next article deals with the question: What programming language should I use for my new project? In "Language Considerations," author Dennis Ludwig looks at some real-world examples of how programming languages were chosen. He then suggests some decision-making parameters that could be formalized into a decision table to aid in the programming language selection process.

In our last theme article, author Richard Riehle clears up some confusion regarding the military's use of Ada. Riehle explains in "SEPR and Programming Language Selection" that the true intent behind abrogating the military's Ada mandate was to make choosing a language part of the Software Engineering Process Review - and not a license to abandon Ada. [...]

Pam Bowers, Managing Editor

From: Richard Riehle
 <richard@adaworks.com>
Date: Sun, 26 Jan 2003 11:12:12 -0800

Organization: AdaWorks Software Engineering
Subject: Ada In Crosstalk
Newsgroups: comp.lang.ada

Check out the current issue of Crosstalk and its support for Ada.

<http://www.stsc.hill.af.mil/crosstalk/2003/02/>

From: mcq95@earthlink.net (Marc A. Criley)

Date: 29 Jan 2003 11:56:36 -0800
Subject: Re: Ada In Crosstalk
Newsgroups: comp.lang.ada

The articles were interesting, discussing "Evolutionary Trends", "Language Considerations", and the original intent of the SEPR (Software Engineering Process Review). It was gratifying to see Ada given a fair hearing in each of these, which is all I, as an advocate of the language, have ever asked for. Good work, Richard, et.al.

Marc A. Criley, Quadrus Corporation, www.quadruscorp.com

Ada Inside

USA - Boeing's C-17 Globemaster III Jet Transport Plane

URL:

<http://www.ghs.com/news/221104b.html>

Green Hills Software Provides Real-Time Operating System For Boeing's C-17 Globemaster III

INTEGRITY-178B RTOS To Fly In C-17 Santa Barbara, CA. November 4, 2002 -- Green Hills Software, Inc., today announced that Boeing has selected its INTEGRITY-178B(TM) real-time operating system (RTOS) for use in the USAF C-17 Globemaster III jet transport plane.

INTEGRITY-178B, along with Green Hills Software's GMART (Green Hills Minimal Ada Run-Time) and GSTART (Green Hills Safe-Tasking Ada Run-Time System), will host the C-17's navigation, mission planning, and display functions. Boeing is also using Green Hills Software's AdaMULTI® Integrated Development Environment (IDE) and G-Cover code coverage tools to develop DO-178B-compliant Ada and C application software for the C-17.

Boeing C-170 The C-17 is a versatile jet transport plane optimized for tactical airlifts, airdrops, and rapid strategic delivery of troops and cargo to main operating and forward bases. With a payload of 160,000 pounds, the C-17 can take off from a 7,600-foot airfield, fly 2,400 nautical miles, land on a small austere airfield in 3,000 feet or less, and be refueled in flight. This combination makes the C-17 ideal for long missions in

rugged terrain, particularly when the take off and landing airfields are small or not well developed.

"Boeing is using Green Hills Software's INTEGRITY and AdaMULTI in the C-17's Avionics System," said Hafez Lorsevedi, Director of C-17 Systems and Software. [...] "Green Hills Software has a long history working with Boeing on projects such as the C-17," said John Carbone, vice president of marketing for Green Hills Software. "INTEGRITY-178B and AdaMULTI are fast becoming the RTOS and IDE of choice for developing and deploying DO-178B-compliant software for safety-critical applications."

INTEGRITY-178B is an ARINC-653 compliant, hard real-time RTOS optimized for safety-critical and mission-critical applications that require the utmost security and fast, predictable response. Utilizing hardware memory protection and an advanced two-level partition scheduler, INTEGRITY-178B provides complete time, space, and resource partitioning between applications operating on the same hardware platform. INTEGRITY-178B also provides guaranteed resource availability in both the time and space domains. This combination facilitates "robust partitioning" (as defined in ARINC 653), enabling applications that have been assigned different DO-178B safety levels to run concurrently on the same processor.

INTEGRITY-178B includes an RTOS simulator (ISIM) that enables programmers to develop and test their code on a PC or workstation without the need for target hardware. INTEGRITY-178B also features a real-time event analyzer (EventAnalyzer(TM)) that enables viewing of system and user events in a graphical display.

INTEGRITY-178B is tightly integrated with Green Hills Software's AdaMULTI® IDE. Together with Green Hills Software's family of optimizing Ada 95, C, and C++ compilers, AdaMULTI automates all aspects of embedded software development, including editing, source-level debugging, program building, run-time error checking, version control, and code/performance optimization. AdaMULTI also features an advanced code coverage tool (GCOVER(TM)) that automates structural coverage analysis for application software, including the analysis of DO-178B Table A-7, Objectives 5, 6, and 7 (statement, decision, and MCDC coverage achievement).

[For information about Green Hills Software, Inc. and contacts, see "Green Hills - INTEGRITY RTOS for Intel x86/Pentium" earlier in this news section. -- dc]

USA - Local Election Results on Saratoga Community Access Television

From: tmoran@acm.org
Date: Wed, 06 Nov 2002 03:14:53 GMT
Subject: A little Ada app
Newsgroups: comp.lang.ada

This evening local election results will be displayed on Saratoga Community Access Television by an Ada program. Saratoga is in Santa Clara County, California, a.k.a. Silicon Valley. The Ada program, using Claw on Windows, will periodically dial in and harvest updated numbers from the county Registrar of Voters web site. The races of local interest will be shown via TV output from the Sony laptop to Saratoga's cable TV subscribers. If all goes well, it will run completely automatically until midnight.

From: tmoran@acm.org
Date: Wed, 06 Nov 2002 18:02:47 GMT
Subject: Re: A little Ada app
Newsgroups: comp.lang.ada

The Ada program and the laptop ran fine. The county Registrar of Voters, OTOH, varied between molasses slow and just plain down. So the Ada program spent a lot of its time patiently trying again and again to get updated numbers. The only glitch was !@#%\$ Pc-cillin popped up at one point asking if this was a convenient time to download the latest virus definitions. Not being entirely trusting of new systems, even when programmed in Ada, I was nearby and squashed that quickly.

France - GHDL, a VHDL Simulator

From: Lionel Draghi
<lionel.draghi@free.fr>
Date: Sun, 01 Dec 2002 01:49:49 +0100
Subject: [ada-france] Une nouvelle appli en Ada : GHDL
To: ada-france@ada-france.org

[Translated from French: -- dc]

The article <http://linuxfr.org/2002/11/30/10480.html> announces the creation of a gcc front-end for the generation of VHDL, which is written in Ada 95. Good luck to GHDL! (<http://ghdl.free.fr/>)

[From that page: -- dc]

GHDL is a VHDL simulator, using the GCC technology. VHDL is a language standardized by IEEE intended for developing electronic systems. GHDL implements the VHDL language according to the IEEE 1076-1987 or IEEE 1076-1993 standard, that is it executes your design. [...]

Technically speaking, GHDL is a GCC front-end: it compiles VHDL files into objects files, like GCC does for C, C++, Fortran or Ada95 files.

GHDL does not generate intermediate C files. GHDL is written in Ada95. I use the GNAT compiler (the GCC Ada compiler) to create the binary image. The GHDL sources includes AGCC, an Ada binding for interfacing with GCC. The GHDL run-time library was also written in Ada95 and uses some of the Ada95 and GNAT packages. [...]

Europe - Ariane 5 Hardware Problem

[Extracts from fr.comp.lang.ada translated from French. -- dc]

From: Xabi Harigoyen
<xabi.harigoyen@netcourrier.com>
Date: Thu, 12 Dec 2002 00:08:35 +0100
Subject: Encore Ada?
Newsgroups: fr.comp.lang.ada

Ariane5 exploded in flight after 3 minutes (motor halted). [...]

From: Preben Randhol
<randhol@pvv.org>
Date: Thu, 12 Dec 2002 12:39:33 +0000 (UTC)

Organization: Norwegian university of science and technology
Subject: Ariane mission failure
Newsgroups: comp.lang.ada

More info here:

<http://www.esa.int/export/esaCP/index.html>

The cause is not found yet. I hope it isn't software related.

From: Stéphane Rivière
<stephane@rochebrune.org>
Date: Tue, 31 Dec 2002 16:33:23 +0100
Subject: Re: Encore Ada?
Newsgroups: fr.comp.lang.ada

The last failure (which relates to an Ariane 5 ESC A, therefore a considerably modified Ariane 5) is due to a leak in the main engine's nozzle's cooling circuit (the central engine, between the two booster rockets). This cooling tube is completely new on an Ariane5 ESC A.

Stéphane Rivière (Ile d'Oléron)

From: Jean-Pierre Rosen
<rosen@adalog.fr>
Date: Wed, 8 Jan 2003 09:56:19 +0100
Subject: Report on failure of Ariane flight 157
Newsgroups: comp.lang.ada

The inquiry board reported yesterday, press release available at http://www.arianespace.com/site/news/03_01_07_release_index.html

In short: The failure was due to a failure of the cooling system of the nozzle of the Vulcain-2 engine. It was the first flight of the Vulcain-2; previous flights used a Vulcain-1. No software involved.

USA - Boeing's B-1B Avionics Upgrade

URL:

<http://www.ghs.com/news/221213b.html>

Boeing Selects INTEGRITY® RTOS for B-1B Avionics Upgrade

Santa Barbara, CA. December 13, 2002 -- Green Hills Software today announced that its INTEGRITY real-time operating system (RTOS) has been selected by the Boeing Company for use in Boeing's B-1B Conventional Mission Upgrade Program (CMUP). The CMUP will add a family of 1760 smart weapons to the B-1B arsenal, and provide the flexibility needed to up-load multiple types of weapons for each mission and launch the appropriate weapon against the selected target. INTEGRITY will be used to enhance the B-1B's onboard avionics flight system so that it can take advantage of CMUP's smart weapons and multiple-target capabilities.

"Boeing is using Green Hills Software's INTEGRITY real-time operating system and AdaMULTI development environment for the avionics, navigation, and weapons systems in the B-1B upgrade program," said Steven Goldman, B-1B avionics manager for Boeing.

"Boeing's selection of INTEGRITY is a terrific honor for Green Hills Software, and we are extremely proud to be contributing to this program," said John Carbone, vice president, marketing at Green Hills Software. Carbone continued, "INTEGRITY has become the RTOS of choice for demanding mission and safety-critical aircraft systems such as those required for the Boeing B-1B CMUP."

To satisfy CMUP requirements, Boeing is upgrading the B-1B's avionics flight system, replacing its six existing computers with four new ones that add significantly more computing power and memory. The avionics flight system provides both offensive and defensive capability, including weapons delivery, radar, terrain following, and navigation.

The computing platform for the new system utilizes a tandem of four computers, each containing a pair of PowerPC processor cards. INTEGRITY provides real-time multitasking, I/O and memory management services for the avionics flight system and hosts the application software that runs on the PowerPC processors.

The developmental flight testing for this new computer system aboard the B-1 has been completed and the Air Force is in the process of performing Operational Testing and Evaluation. That is to be completed in late 2002. Additionally, the Air Force has contracted for the procurement of the retrofit kits necessary to upgrade the B-1 fleet.

INTEGRITY is a fast, deterministic, RTOS designed for applications that require high reliability, availability, security and a cost-effective business model. Utilizing the hardware memory protection facilities of the processor's MMU, INTEGRITY builds a firewall between the kernel and user tasks that prevents errant or malicious tasks from corrupting user data, the kernel, interprocess communications, device drivers and other user tasks. INTEGRITY also enhances reliability and determinism by running with interrupts continuously enabled and guaranteeing access to the CPU and memory for critical tasks. Other RTOS solutions do not offer this protection or security, leaving systems vulnerable to failure from errant application, viruses, and hackers.

[For information about Green Hills Software, Inc. and contacts, see "Green Hills - INTEGRITY RTOS for Intel x86/Pentium" earlier in this news section. -- dc]

From: Harbaugh, John S

<john.s.harbaugh2@boeing.com>

Date: Fri, 17 Jan 2003 08:51:37 -0800

Subject: Re: B-1B Upgrade

To: team-ada@acm.org

Boeing is wrapping up it's portion of the Smart Bomb Rack Controller upgrade. The design is highly concurrent, and uses Ravenscar principles of cyclic tasks interacting via protected objects. The use of deadline monotonic scheduling and Ravenscar design principles yielded a solid design with predictable behavior, and on-time delivery (well there was some schedule slide due to our failure to plan on the earthquake two years ago.) ;-)

Australia - FedSat-1 Microsatellite Launched

From: Alan and Carmel Brain

<aebrain@webone.com.au>

Date: Thu, 19 Dec 2002 16:39:12 +1100

Subject: FedSat - Ada Inside

To: team-ada@acm.org

On Saturday 14th December 2002, a Japanese H-2A booster launched FedSat, Australia's first satellite for 30 years - and the first foreign payload launched with the H-2A.

RealPlayer Report:

http://www.abc.net.au/news/2002/12/14/video/20021214pm_space.ram

Video and Stills of launch and payload separation are at

http://h2a.nasda.go.jp/live_e.html

Article about the software and hardware on board

<http://www.theage.com.au/articles/2002/08/24/1030052995897.html>

Rogue's Gallery of the engineering team

<http://www.crcss.csiro.au/spin/spin90/spin9007.html>

Lastest report:

http://abc.net.au/news/scitech/2002/12/item20021218103335_1.htm

[See also "Australia - Spaceflight Avionics Software on FedSat-1 Microsatellite" in AUJ 23-1 (Mar 2002), pp.34-35. -- dc]

This microsatellite is possibly the densest in terms of payloads-per-cubic metre and payloads-per-kilo that has yet been orbited. NewMag Magnetometer, GPS Ionospheric probe, Field Programmable Gate Array experiment, and the Communications payload with 2 different experimental comms systems and an experimental processor, plus a star camera. All from different organisations, with different computers, some big-endian, some little-endian, some with 1750A floating-point format, some with IEEE, and using different languages. In a cube that's about 1/2metre on a side, massing just 58 kg. All up cost less than \$200,000 US (\$400,000 Aus) per Kg.

The on-board computer that is "one ring to rule them all" - an ERC-32 running at a glacial 8 MHz, with 16 MB of normal RAM. This has to communicate with all the payloads, store their data in mass memory, plus monitor hundreds of different voltages, temperatures, comms statuses, reprogram the payload computers as required, start and stop experimental campaigns out of ground contact, and relay many Megabytes of telemetry in the 15-20 mins per day its in contact. Oh yes, it can be reprogrammed from the ground too while in flight.

It's the first satellite we've built in 30 years, so was built by a team still "learning on the job" with no outside help.

The language on the On Board Computer? Ada-95. Of course. The compiler? Gnat 3.13p. *That* is why we got contact in the first ground pass - not just contact, but (3,25) housekeeping parameter report messages, even during tumbling and periodic blackouts. And were able to activate the ACS (Attitude Control System) nominally at first attempt.

"In Space, no-one can press CTRL-ALT-DEL". But with Ada, you don't have to.

Alan Brain (Head of FedSat On-Board Computer Software Development)

USA - AdaIC's E-mail Spam Filtering

From: Randy Brukardt

<randy@rrsoftware.com>

Date: Mon, 6 Jan 2003 13:23:36 -0600

Subject: Re: [off-topic] spam-filters Newsgroups: comp.lang.ada

Actually, most spam these days are short HTML messages with little or no text. There isn't much load on the server from

them. Simply blocking HTML graphics gets rid of many of them. Also, a lot of spam is now encoded in various ways so that simple text filters can't find them. The big messages tend to be real or are viruses. (I'm seeing 3-5 viruses a day in the various filters; most are sent to the public webmaster and mailing list addresses we support.)

I know this well, because I wrote a spam-filtering plug-in for our mailserver (in Ada, of course). Our mail server is receiving about 150 messages per day, of which the filter passes only about 15%. About 30% are so frequent and obvious that I'm able to match them and autodelete them. And many of those get caught by the (old) blacklist filtering, which runs after my filter.

[And from a later message: -- dc]

In any case, multiple layers of defense are the only option. And the more of those layers that are written in Ada or other reliable programming languages, the more likely that they actually will work and not cause additional problems.

USA - AdaIC's Archive Web Server

From: Randy Brukaradt
<randy@rrsoftware.com>

Date: Mon, 6 Jan 2003 14:47:57 -0600

Subject: Re: Anybody in US using Ada
Newsgroups: comp.lang.ada

> If you make a new web server, it still has to interoperate with most of the things other webservers work with, [...]

If you "interoperate" with those other things, (i.e. plugins and CGIs), you're also bringing in the intractable security problems of those other things. That's precisely why I updated Tom Moran's web server to use for our AdaIC backup server. It doesn't know how to execute another program (and never will), so no one will ever be able to use it to launch Cmd.Exe, no matter what garbage it is given. And most of the other security problems you hear about can't happen, either. The only thing it writes is log files, so an attacker can't use it to create files, either, no matter what they do. About the worst that can happen is that they could use it to cause a denial-of-service -- and given that it uses a set of Ada tasks, even that would be fairly difficult.

Of course, it has to understand HTTP and make log files that analyzer programs recognize, but the first is the definition of "web server" and the latter is trivial.

But, all of that said, I think your basic point is correct.

[See also "USA / AdaIC - Secondary Web Site" in AUJ 23-1 (Mar 2002), p.34. -- dc]

From: Randy Brukaradt
<randy@rrsoftware.com>

Date: Tue, 7 Jan 2003 13:05:34 -0600

Subject: Re: Anybody in US using Ada
Newsgroups: comp.lang.ada

> Seriously, I think an Ada webserver is a good thing, but unless you sell it from the security standpoint (which _severely_ limits its audience) it's not something "new".

Well, security is where it's at when it comes to servers on the Internet. People make millions describing how to secure servers. Certainly, it helps to not have software running on it that has unnecessary security holes.

But all that said, it would be hard to market YAWS (Yet Another Web Server), even if it was highly secure. It would also have to have better performance than anything built directly on top of Windows sockets can. So, I don't currently have any intention to market or even make available the web server I wrote. [...]

Indirect Information on Ada Usage

[Extracts from and translations of job-ads and other postings illustrating Ada usage around the world. -- dc]

From: Alan and Carmel Brain
<aebrain@webone.com.au>

Date: Wed, 30 Oct 2002 22:45:28 +1100

Subject: Avoiding Kludges
Newsgroups: comp.lang.ada

[...] I've got a dozen successful Ada-83 projects under my belt over nearly 20 years, and was in charge of the Ada-95 spaceflight avionics for FedSat, due for launch in December. [...]

From: Richard Riehle
<richard@adaworks.com>

Date: Tue, 05 Nov 2002 20:49:28 -0800

Organization: AdaWorks Software
Engineering

Subject: Request for an Ada Programmer
Newsgroups: comp.lang.ada

[Forwarded by Richard Riehle to comp.lang.ada upon request. -- dc]

I am looking for an Ada programmer to finish a project that I have been working on for about three years. This is a networked video bingo system for commercial use written in Ada and the OS is DR DOS. We chose these for integrity and stability of the system.

The programmer was my husband who had a stroke two years ago and has not finished the software. The first useable version is very close to ready, with ongoing support needed. Do you know of an Ada programmer in or near Wyoming that would be interested in this project?

Marylin M Sutton, Romar Technology,
marylins@vcn.com

From: Kees de LezenneCoulander
<lezenne@compuserve.com>

Date: Thu, 7 Nov 2002 14:05:21 -0500

Subject: Re: Can I treat Current_Output as a file of bytes?

Newsgroups: comp.lang.ada

We have technical calculation programs that read input from fairly large and complicated ASCII files. I find it quite a nice touch that I can quote page, line and column number when an error occurs.

C.M. de Lezenne Coulander, Aircraft Development and Systems Engineering B.V., Hoofddorp, The Netherlands

From: Peter Atterfjäll
<peter@atterfjall.pp.se>

Date: Sat, 09 Nov 2002 08:53:41 +0100

Subject: gnat 3.14 RAW Socket support?
Newsgroups: comp.lang.ada

Last year I wrote a telecom application using gnat 3.13 on top of Red Hat 7.0. Since I had really limited socket needs I implemented some thin bindings myself that supported the Packet_Socket. The whole experience with Packet or Raw socket was however it was very poor documented for Linux. In the latest 3.14 release of gnat I have seen there is built in Socket support. [...] This is only a 5000 line hobby project, but it would feel good to make the code a bit more portable than what I believe it to be today, [...]

From: crazy_diver@hotmail.com
(crazy_diver)

Date: 25 Nov 2002 08:02:58 -0800

Subject: ada et gestion du port serie
Newsgroups: fr.comp.lang.ada

I'm currently writing a program in Ada to recover the data recorded by a "dive computer" to process it on a PC running Windows 98. [...]

From: Richard Riehle
<richard@adaworks.com>

Date: Sun, 01 Dec 2002 11:02:24 -0800

Organization: AdaWorks Software
Engineering

Subject: Re: New Ada compiler for .NET
Newsgroups: comp.lang.ada

[...] A couple of years ago we received a request for Ada training from Iran. Every now and then, we get a request from some other potential enemy. We don't do training for these countries, but the fact that they are doing projects in Ada is amusing. My little booklet, Ada Distilled, has been downloaded by people from all over the world including many places in the Middle East. GNAT compilers are also being downloaded from a lot of places we don't have particularly friendly relations. [...]

From: Richard Riehle
<richard@adaworks.com>

Date: Thu, 09 Jan 2003 18:43:56 -0800

Organization: AdaWorks Software
Engineering

Subject: Re: Hijacking a Thread
Newsgroups: comp.lang.ada

[Ada] is also used in Iran, Russia, Iraq, North Korea, and several other places one would not ordinarily consider as Western

countries. I get requests for training from some of these places (not North Korea or Iraq, however). We don't do business with Iran either. It is also used for some very large, and I mean very large projects in China. Of course the Chinese have developed their own Ada compilers and I think they have modified them in some ways. One Chinese innovation I saw involved using a construct similar to a typical C++ class, but without all the associated C++ garbage code. I'll try to find an electronic copy of one of those specifications and post it in this forum.

From: Adrian Hoe
<mailbox@adrianhoe.com>
Date: Fri, 10 Jan 2003 13:17:13 +0800
Subject: Re: Hijacking a Thread
Newsgroups: comp.lang.ada

> From my sources, Iran is using Ada extensively. So do Pakistan and India (of course). North Korea obtains Ada knowledge from South Korea. South Korea uses Ada extensively in its transportation systems, power plants, some military installations. Don't bother to ask me my source. It's reliable.

[From another message: -- dc]

Tsinghua University (Beijing) has had some research on Ada some time ago. If my memory is not corrupted, so is Inner-Mongolia.

From: David Carlson
<david_carlson@oxfordcorp.com>
Date: Mon, 9 Dec 2002 10:58:45 -0500
Organization: Oxford Global Resources, INC
Subject: assistance
To: team-ada@acm.org

[...] I have a client looking for a strong Ada background who has worked on Air Traffic Management systems. The work is being done for a client in the mid Atlantic. This is a contract position. [...]

From: Ada Marketing
<adamark@sd.aonix.com>
Date: Sat, 14 Dec 2002 09:49:56 -0500
Subject: Re: JSF and C
To: team-ada@acm.org

> I thought they had picked Greenhills Ada & RT (royalty free) for the JSF? [Joint Strike Fighter -- dc]

I think this is true for part of the aircraft. Pratt and Whitney has selected Aonix's ObjectAda/Raven for the jet engines.

From: Marin David Condic
<mcondic@acm.org>
Date: Tue, 31 Dec 2002 22:20:26 -0500
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

I believe Pratt & Whitney is still using Ada for engine controls. At least some of them. As one who is working on the engines for the JSF, I can tell you that those controls are still in Ada [...]

From: John R. Strohm
<strohm@airmail.net>

Date: Tue, 31 Dec 2002 12:55:33 -0600
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

I *think* that Lockheed-Martin Fort Worth is using Ada for F-22 and F-16. You might try looking at their web sight (<http://www.lmco.com>) and seeing what they are looking to hire. [...]

From: britt@acm.org (Britt Snodgrass)
Date: 31 Dec 2002 14:38:07 -0800
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

Rockwell Collins still uses Ada 83 or Ada 95 in many new avionics products. Many products use a mixture of Ada and C.

From: Dennis Lee Biebr
<wlfraed@ix.netcom.com>
Date: Tue, 31 Dec 2002 15:54:35 -0800
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

LM Sunnyvale, Air-borne Laser, also has Ada mentioned in some of the openings [...]

From: prichtmyer@yahoo.com (Peter Richtmyer)
Date: 4 Jan 2003 18:52:19 -0800
Subject: Ada and VxWorks VxMP
Newsgroups: comp.lang.ada

We will be using VxWorks VxMP Shared Memory Objects between two boards, our code is in Ada. We will be passing messages using Shared Message Queues, [...]

From: Vincent Hoefler
<JeLlyFish.software@gmx.net>
Date: Sun, 05 Jan 2003 15:38:36 -0500
Organization: JeLlyFish software
Subject: Re: advantages or disadvantages of Ada over Pascal or Modula
Newsgroups: comp.lang.ada

The EADS uses Ada and I am not sure, but I think, a lot of the ICE related software is written in Ada, too.

From: Stéphane Rivière
<stephane@rochebrune.org>
Date: Tue, 07 Jan 2003 19:20:39 +0100
Subject: Re: [ada-france] Ada et le graphique
cc: Ada France <ada-france@ada-france.org>

Approximately two years ago I spent quite some time [working on graphical applications] with as most important criteria: multi-platform; "pure" Ada solution [...]; reliable and really portable; with prospects for evolution [...]. We have developed several applications under Windows. To port them to Linux was child's play. IIRC, modifying one Gtk call for all applications. Science fiction :) [...] We have been very satisfied with this experience. And we will start again...

Stéphane (Ile d'Oléron)
Date: Thu, 16 Jan 2003 11:53:11 +0100 (MET)
Subject: Posted on monster.be website

Belgium, Vlaams-Brabant, Software engineer Ada, FullTime, Permanent

[...] The more experienced applicants will be responsible for Object Oriented Analysis review, the architecture and design of applications, with special care for availability, maintainability, performance and reliability. The missions will also involve analysis, software design, documentation and testing. The consultants will be involved in the validation of the design, the performances and the reliability. [...]

Profile: several years of experience in the specification and design of application software, preferably in a critical environment; sound experience in the production of technical specifications for systems; very good knowledge of the programming language Ada; good knowledge of Unix and NT Platforms; good knowledge of CORBA and/or UML and/or Oracle are advantages. [...]

From: 402450@cepsz.unizar.es (Jano)
Date: 17 Jan 2003 03:03:13 -0800
Subject: Best option for portable GUI?
Newsgroups: comp.lang.ada

I'm near to start a project of some size using Ada. I intend to make it portable, at least between Windows and Linux. Part of this project is a GUI. [...] I guess there are some libraries for GUIs in Ada that would help me in my objective of portability. [...]

From: Roger Racine
<rracine@draper.com>
Date: Fri, 17 Jan 2003 09:36:14 -0500
Subject: Ada -- Increase or Decrease to Risk?
To: team-ada@acm.org

I am the software task lead for a new project. I (of course) selected Ada as the language, to decrease the cost and schedule risk for this part of the project. Unfortunately, as soon as the top three people (Program Manager, Technical Director, and a systems engineer) at my company found out about it, they immediately thought I had increased the risk, and asked for justification for using what was thought to be unproven technology.

I got by the Technical Director without much problem by pointing him to some other similar programs that used Ada. He also checked with one of our consultants, who had no problem with our using Ada.

However, it turns out the Program Manager and the systems engineer had both had problems with Ada in the one program each of them were on (early 90s) that used it. It further turns out that both programs (different companies) used the same compiler vendor, and their real problem was with the compiler, not the language. [...]

After pointing out that A) that compiler vendor was one out of many; and B) Ada

95 implementation caused some major changes to the way vendors made compilers; and then after getting some information from the compiler vendor I must use (different story), I think I have relieved their concerns to some extent.

I don't know what to do about this in general. Somehow, people need to be educated that there have always been, and always will be, bad tools out there, but that does not mean that the technology is bad.

It did help to show that the front end was used by many compiler vendors (and all targets for the required vendor's Ada compilers) and the back end is used by the vendor's C compiler (which is well trusted). [...]

[Stephen Leake
<Stephen.A.Leake@nasa.gov>
responded: -- dc]

Sounds like you did a good job. You listened to their concerns, and they listened to your answers. Now just have a successful project, and you'll have more Ada converts!

*From: karl bowl <karl.bowl@gmx.de>
Date: Mon, 20 Jan 2003 16:56:04 +0100
Subject: Re: advantages or disadvantages of
Ada over Pascal or Modula
Newsgroups: comp.lang.ada*

I belong to a group of programmers for movement control systems based on 80x86-Processor (embedded system). As far as now, we have used Pascal 86. But now, we would like to change, because the tools for sw-development in this language have been going rare.

There are a group of colleagues, they prefer Modula2. But I think, Modula2 is not used very often and so there are no distributions, which provide good tools for sw-development (i.e. a source code debugger for the target system). Therefore, I am looking for alternatives - i. e. Ada. Another big point is, that the language we choose, should be a language with strong type checking. The price is important for us but less important as the point mentioned before.

*From: Jerry Petrey
<jdpetrey@raytheon.com>
Date: Mon, 20 Jan 2003 10:39:47 -0700
Organization: Raytheon Company
Subject: Re: advantages or disadvantages of
Ada over Pascal or Modula
Newsgroups: comp.lang.ada*

We use the DDC-I Ada compiler for a 80x86 target in an embedded system and have been very happy with their tools and support. Take a look at their new SCORE product (<http://www.ddci.com/>). I think you are right in considering Ada.

Jerry Petrey, Senior Principal Systems Engineer,
Raytheon Missile Systems

*From: Frank Piron
<frank.piron@konad.de>
Date: Thu, 23 Jan 2003 15:07:08 +0000
Organization: KonAd GmbH
Subject: GtkAda for professional GUI-
Application?
Newsgroups: comp.lang.ada*

Hi to all, who want to spread Ada in Germany. In the next two years we will develop a client library for a production Workflow-System. Since we wrote most of the Server-Code in ORACLE PL/SQL, which is similar to Ada83, we begun developing a Client-Side Library in Ada95. Part of this Library will also be a GUI. [...]

*From: Adrian Knoth
<adi@drcomp.erfurt.thur.de>
Date: 23 Jan 2003 15:17:51 GMT
Organization: Modern Electronics
Subject: Re: GtkAda for professional GUI-
Application?
Newsgroups: comp.lang.ada*

[...] I (we) have built a centralized-administration-poolsystem for running a lot of Linux-Nodes without the need to maintain every single PC. In combination this would be an efficient way for companies to handle an enterprise-wide Linux-installation. One Admin for all, one installation for all, but speed, security (Kerberos V, AFS) and extensibility (adding a new PC takes less than five minutes, then it boots right into X).

*From: Jerry Petrey
<jdpetrey@raytheon.com>
Date: Fri, 24 Jan 2003 08:18:14 -0700
Organization: Raytheon Company
Subject: Re: Serial port programming.
Newsgroups: comp.lang.ada*

I have recently used Stephen Leake's com port package to do a serial port driver for a hardware simulator GUI application running on Windows and using GNAT Ada. [...]

*From: Stefan Soos <stefan.soos@gmx.de>
Date: Wed, 29 Jan 2003 11:09:40 +0100
Subject: AdaSDL and gTexImage2D
Newsgroups: comp.lang.ada*

I'm doing some programming in Ada with the thin binding to the SDL library. I'd like to create some textured objects, [...]

*Date: Thu, 30 Jan 2003 11:09:14 +0100
(MET)
Subject: Posted on job-scareer.be website
New! 4 Ada Software Engineers [...]*

You will be integrated in a strategic project including architectural & detailed design, development of the application, programming, testing and writing of the design documentation. [...] Knowledge of a programming language preferably Ada 83-95 or C++. [...]

*From: Alexander Boucke
<alex@lufmech.rwth-aachen.de>
Date: Fri, 31 Jan 2003 11:17:08 +0100*

*Organization: Lehr- und Forschungsgebiet
f. Mechanik
Subject: Re: Gnat:
system__finalization_implementation__fi
nalize_list
Newsgroups: comp.lang.ada*

[Problem description removed. -- dc] I've had exactly the same problem [...] Otherwise I am very pleased with Ada and gnat, looking at a colleague and his C++ struggles :-)

*From: Thierry Poulain
<thierry.poulain@univ-valenciennes.fr>
Date: Mon, 3 Feb 2003 16:04:23 +0100
To: <ada-france@ada-france.org>*

I recently ported an application written initially in Ada83 under an OpenVMS environment to Linux (Mandrake 9.0, Bi-processor AMD PC). For that I installed Gnat 3.15. The application in question represents a substantial amount of code (among others an air traffic simulator, support tools, data servers, ...). It functions perfectly, [...]

*From: mcq95@earthlink.net (Marc A.
Criley)
Date: 5 Feb 2003 08:59:00 -0800
Subject: ADV: Quadrus Hiring
System/SW/Test Engineers
Newsgroups: comp.lang.ada*

[...] looking for system, software, and test engineers [...]

Our primary focus right now is for system and test engineers, especially those who have experience with national and theater missile defense systems (e.g., NMD, THAAD). However, we are looking for some SW engineers as well, and Ada is well looked upon here. At the moment there is only a modest amount of software development occurring, but some of that internal development is being done in Ada. [...]

Thoughts on the Future of Ada

*From: Peter Amey <peter.amey@praxis-
cs.co.uk>
Date: Tue, 3 Dec 2002 15:17:39 -0000
Subject: Re: Future of Ada
To: team-ada@acm.org*

[On a company that seems to consider transitioning from Ada to other languages (e.g., C, C++) although they have lots of working Ada code, but they claim to perceive that the supporting infrastructure (e.g., skilled labor, tools) is dwindling. -- dc]

My thoughts are that we should continue, energetically, to explain why their migration away from Ada is wrong-headed, based on false premises and likely to be an expensive mistake.

We have some good evidence that Ada (and SPARK) can reduce cost and raise quality. On the C130J programme, the error density in C code was an order of

magnitude higher than that of the Ada and two orders of magnitude higher than that of SPARK. This is for code that was already certified by the FAA to DO-178B level A and which had all been produced by skilled practitioners with good tools. Furthermore, the SPARK code (100 times better remember) cost a quarter of Lockheed's usual cost for Level A.

We should be asking these engineering professionals what overwhelming, unavoidable force is compelling them to accept one to two orders of magnitude higher error rates together with increased costs. There reasons must be truly impressive to make such a backward step.

Tool support is a red herring. Most of the Ada compiler vendors use compiler back ends that are common to other languages. So we are to believe that Greenhills C++ will be available for ever but Ada is dead? GCC is eternal but GNAT is doomed? Why don't we ever hear people complaining about moving their legacy Whitesmiths or K&R C to a new platform for which there is only C90 available (I'd say C99, but of course there aren't any C99 compilers)? This is a common and real problem that they just put up with.

As for skilled people, the perceived shortage is due to the bizarre way our industry confuses skills and products. The most important quality driver in software systems development is domain expertise. It is no good having millions of C++ programmers available if you can't trust them to build your flight control system. If an engineer is safe to be let loose on such a control system, I can teach him SPARK in a week. If he is not fit to do so then I can't teach him to be a good engineer in a lifetime (even if he is the best C++ hacker in the world). At Praxis, we solve the skill shortage by recruiting good engineers (even if they don't have that common non sequitur: "10 years experience of the latest fad") and teaching them the languages we need to use. We don't allow our engineering judgements to be dictated by what is currently "cool".

What is clear is that if professional engineers keep selecting second best solutions for spurious reasons then they will eventually only have second best solutions to choose from. They will have been proved right, of course, Ada will have died, but it will be a Pyrrhic victory. Anyone fancy writing a flight control computer in Visual C# for embedded WinCE? Thought not.

Peter

P.S. We are currently helping a client replace a Jovial system with a brand, spanking new Ada one!

*From: Jacob Sparre Andersen
<sparre@nbi.dk>*

*Date: Tue, 3 Dec 2002 20:45:22 +0100
Subject: Re: Future of Ada
To: team-ada@acm.org*

I would like to start by saying that from what I can see here in Copenhagen right now, Ada has a reasonably bright and growing future [...] An acquaintance and I have recently managed to start an Ada SIG in the local Linux User Group, SSLUG, with (reasonably) regular meetings and continuously growing interest (but maybe it simply helps to start from a low level). [...]

> I am sure Ada can sell well, because reliability of software can sell. People have had enough of BSODs of different sorts already. [...]

Some people are as far as I can see slowly getting the point.

Ada and Uncertain or Changing Requirements

*From: Bodily, Sue <sbodil@sandia.gov>
Date: Fri, 6 Dec 2002 10:18:57 -0700
Subject: Re: Future of Ada
To: team-ada@acm.org*

The environment that you mention (uncertain and/or changing requirements) is one that I work in every day. I have successfully and productively used Ada in that environment and find it much superior to C++, which I have also used. [...]

Susan Bodily, Sandia National Labs

*From: John P. Woodruff
<WoodruffJ@llnl.gov>
Date: Fri, 6 Dec 2002 15:19:51 -0800
Subject: Re: Future of Ada
To: team-ada@acm.org*

I like to confirm the writer's sense of value of Ada when requirements are not fully cooked. A report that I prepared for next week's SigAda conference remarks:

"The context of development of a large experimental facility such as NIF has distinct challenges for software engineers. It is not possible to construct a complete set of requirements a priori: the facility itself is being designed concurrently with the control system that will operate it. This observation applies at every level of the organization of the software system.

At the hardware interface, numerous components of the laser are being invented as the controls for those components are being constructed, so only during integration testing of first units does the software developer refine the details of device control. The same condition occurs in large-scale integration. Since the techniques for operating the laser are refined while the first beamlines are being activated, high-level requirements emerge as preliminary operations elaborate the conditions for successful exploitation of the innovative design."

I think the presentation (next Tuesday) by my colleague Bob Carey on the operational experience using Ada in the

National Ignition Facility will encourage Ada-philes of the value of Ada in our very large controls project.

*From: David Botton <David@Botton.com>
Date: Sat, 7 Dec 2002 00:16:02 -0500
Subject: Re: Future of Ada
To: team-ada@acm.org*

I use Ada for the same purposes as you use C++. Being fluent in both dialects, I find that use of Ada is a superior choice for prototyping and early design thanks to (but not only thanks to) its higher level language support for tasking, flexible partitioning both with in and out of the OO paradigm of code and exception handling (of course C++ does support this particular feature). Even when required to develop a project in C++, I usually write the prototypes in Ada and all the test code in Ada. This way when I write the C++ version it looks like Ada code, designed not hacked :-)

The strong typing not only doesn't prevent me from making progress, but it insures that when I make a change to the class/type it doesn't slip by the compiler.

While true, C++ was designed for fun (so says its creator) and Ada for engineering, I find that Ada works equally well for "playing around" or XP (as in Extreme, not the broken OS) style development.

Be as Visible as Possible

*From: Roger Racine
<rracine@draper.com>
Date: Fri, 6 Dec 2002 12:50:21 -0500
Subject: Re: Future of Ada
To: team-ada@acm.org*

[...] One other thing (maybe more practical, but related):

Be as visible as possible. My manager went to the Embedded Systems Conference in Boston last month, and came back saying she heard absolutely no mention of Ada. I went to the Digital Avionics Systems Conference and can say the same thing. It does not matter if Ada is used by only 1 percent of developers; if the perception can be made that it is larger than it is, it might have a chance of growing. If, on the other hand, it appears that no one is using Ada (the impression left on my manager), it is more likely to go in the other direction.

By the way, I just went through the exercise of convincing the Program Manager and Technical Director, on a project just starting, to allow me to use Ada. It was not easy, because of the above perception. Luckily, I was able to point to some web sites (thanks to the web site maintained by Michael Feldman) showing that Ada is being used on similar programs (this program is not quite under contract yet, so I will not mention its name).

Roger Racine, Draper Laboratory, Cambridge, MA, USA

From: Luke (Lujun) Zhang
<lujunz@yahoo.com>
Date: Sat, 18 Jan 2003 15:38:57 -0800
Subject: Re: Ada -- Increase or Decrease to Risk?
To: team-ada@acm.org

> I guess I would ask as many people as possible to write up success stories like the one at <http://www.ghs.com/wp/vis2.html>.

There are a lot of successful stories you might use, for example,
<http://archive.adaic.com/projects/successes.html>

From: Randy Brukardt
<Randy@rrsoftware.com>
Date: Fri, 17 Jan 2003 13:44:10 -0600
Subject: Re: Ada -- Increase or Decrease to Risk?
To: team-ada@acm.org

The AdaIC is always looking for success stories, and will help in the writing of them. So if you have any good stories and don't have time to write them up, contact Ann Brandon at webmaster@adaic.com.

Randy Brukardt, Technical Webmaster,
adaic.org

Ada in Context

On Readability vs. Writability

From: Richard Riehle
<richard@adaworks.com>
Date: Sun, 17 Nov 2002 17:25:51 -0800
Organization: AdaWorks Software Engineering
Subject: Re: Let's change semantics of "use type"
Newsgroups: comp.lang.ada

We should be cautious about adding features to the Ada language that will dilute its essential power. We should be wary of trying to make Ada look like something else in pursuit of a potentially ephemeral popularity. Ada is what it is: a language designed to enable the compiler to detect the maximum number of errors as early in the development process as possible and to continue catching errors during execution and over the full life-cycle of the maintenance process.

Since Dr. Dewar has chosen to suspend his contributory edification to comp.lang.ada for a while, it is probably worthwhile to recall one of his most memorable and most important admonishments about Ada versus other programming languages.

"Ada is meant to be more readable than writeable."

A corollary to this is that Ada is intended to be more traceable than writeable. Anyone who has had the entertaining experience of slogging through pages of C or C++ code in search of (yes, I know

there are tools for this) some function or variable knows the importance of this. [...]

From: Richard Riehle
<richard@adaworks.com>
Date: Sun, 15 Dec 2002 23:19:57 -0800
Organization: AdaWorks Software Engineering
Subject: Re: IBM Acquires Rational Ada Newsgroups: comp.lang.ada

[...] One of my colleagues asked me recently why I was so reluctant to use C or C++ for my programs. I replied that C++ was reminiscent of the strike-anywhere matches our grandpappies named "barnburners."

Just today, I was reading a book on software architecture in which the authors acknowledged that the vast majority of C++ is noted for being unmaintainable by anyone except its creator.

The fact that C and C++ is so widely used to create popular software is no different than the fact that so many people select MacDonald's, Burger King, or Kentucky Fried Mynah Bird, for their sustenance under the illusion they are actually being nourished. The fact that something is popular does not make it good.

I am required, during these past couple of years, to spend more and more time with C++. The more time I spend with it, the more horrified I am at the thought it is being used for our military weapon systems.

If C++ is the best we can do, this industry is in trouble for a very long time to come.

Productivity is More than Speed of Coding

From: Marin David Condit
<mcondic@acm.org>
Date: Thu, 12 Dec 2002 08:20:19 -0500
Subject: Re: IBM Acquires Rational Ada Newsgroups: comp.lang.ada

Hyman Rosen <hyrosen@mail.com> wrote

> You know, no one loves C and C++ more than I do, but I don't think I would be any slower coding in Ada than in either of the above languages (once I got a good working knowledge of the language, of course). [...]

The metrics I've collected in the past didn't have anything to do with how fast someone could code-up some routine. In that respect, Ada probably wouldn't have any advantage over C or C++ - possibly even a disadvantage. What my metrics demonstrated was that when the whole development cycle was complete (requirements through formal testing) using Ada meant a reduction in man-hours spent (by about 50%) and a reduction in bugs found (by a factor of 4). My study was not alone. There was also the famous Ada/Model-Railroad study that demonstrated an impressive improvement

in productivity and reduction in errors in a similar environment. (realtime controls)

[See also "On Languages, Productivity and Quality" in AUJ 22-4 (Dec 2001), pp.222-224, and "Measuring Productivity" in AUJ 22-3 (Sep 2001), p.167. -- dc]

I wouldn't claim that you couldn't code just as fast - or even faster than I can code, given our favorite languages. I'll even go so far as to say that you'd be in that "Any Competent C Programmer" category who never makes stupid mistakes in coding that cost untold hours down the line in debugging. But given a random selection of developers with varying levels of skills and the need to work together on a large development project, you've got a different animal on your hands. In that situation, there is data to indicate that when other factors are equal, (id est, nobody gets huge libraries of utilities or other leverage) Ada is more productive and results in fewer errors over the development cycle.

From: tmoran@acm.org
Date: Wed, 08 Jan 2003 18:04:18 GMT
Subject: Re: Anybody in US using Ada? Newsgroups: comp.lang.ada

Personally, I'm periodically amazed to find how much faster it is to get an Ada program running right than with previous languages I've used. "Once it compiles in Ada, it's close to being done." Or perhaps I'm just a whole lot smarter than I used to be. ;)

On Languages and Maintenance Cost

From: David Emery <demery@cox.net>
Date: Fri, 03 Jan 2003 00:41:13 GMT
Subject: Re: Anybody in US using Ada? Newsgroups: comp.lang.ada

In 1990 MITRE did an internal study based on then available data saying that the cost to maintain Ada was -LINEAR- on SLOC, not exponential as Barry Boehm says. If you think about this, it's a radical conclusion. And it wasn't done by the Ada people at MITRE, but by our cost center's statisticians and cost analysts.

Back when Emmett Paige was ASD-C31, he held a series of Ada Dual-Use summits. At each, my position was: DoD has the data available to show if Ada has the value we claim. Let's collect the data and let the facts speak for themselves.

Two results from this experience: 1. DoD didn't want to gather and then hear facts. 2. Language decisions are generally not made on the basis of -any facts- but rather management perception of "acceptability", "training costs", etc.

Dave (been there, done that, forgot my T-shirt)

On Early Adopters and Productivity

From: John R. Strohm
<strohm@airmail.net>

Date: Fri, 3 Jan 2003 12:40:27 -0600
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

> Those were big companies and the projects were very long lived where software development was not going to be the cost or schedule driver. How many commercial products are in that position? [...]

I don't know whether I can use this as a counterexample or not. My recollection is that Silicon Graphics was using Ada internally VERY early in their life, and they were keeping this VERY quiet because it was helping them, a lot, and they didn't really want anyone else to know.

Rational used Ada internally, and it helped them a lot. They published their numbers, but the reaction at GD/FW was that it just wasn't possible to get those kinds of numbers.

McDonnell-Douglas had been using assembly language on F-15. For the IFFC/Firefly demonstration, they jumped into Ada with both feet, enthusiastically, and reported very good numbers, for a digital flight control application. (IFFC/Firefly was Integrated Flight and Fire Controls: the idea was to let the firecontrol computers cue the flight control system directly, to let the airplane help the pilot point the airplane and the weapons at the target. The demo pilot reportedly said, very enthusiastically, "I don't know if I fired the gun or the airplane did, but we got him!")

I do know that Silicon Graphics was unique among workstation manufacturers for having a solid Ada toolset long before anyone else did. [...]

From: John R. Strohm
<strohm@airmail.net>

Date: Sat, 4 Jan 2003 10:34:39 -0600
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

[About Rational: -- dc]

A few reports trickled out that other people using the R1000 were getting similar numbers.

Also, it is very easy to reality-check the Rational numbers, to a rough order of magnitude: take the number of SLOC in the product, which they reported, divide by the number of years the company had been in existence, and you get SLOC/year. From there, it is utterly trivial to divide by trial values of SLOC/man/year to estimate the number of programmers involved, and then compare them to the company headcount and money burn rate.

The short answer is that Rational's numbers stand up under reality checking.

[About McDonnell-Douglas: -- dc]

Their numbers were good even for high-order language development.

There is this, also. At the time, the conventional wisdom was that Ada would not EVER be suitable for "real" embedded work. Using it for digital flight controls *and* firecontrol, and doing a successful demonstration (gun shootdown of a QF-102 drone from a 90 degree aspect: something that is essentially impossible for a human pilot on his own), *AND* getting good productivity numbers to boot, was a very solid counter to that belief.

Ada Ports Command and Control System to MS Windows NT & Pentium

URL:

<http://www.adaic.org/news/adarose.html>
Subject: AdaIC Feature Story [...]

[Regardless of the application domain, this feature story provides interesting information on the easy portability that can be achieved with Ada software. -- dc]

Ada Software Ports Easily to MS NT and Pentium

Ada95 Runs Realtime Command & Control System on Windows OS & Pentium

By Ann S. Brandon

One of the promises of Ada is portability. Yet managers of legacy code in Fortran, SmallTalk, or Ada often assume that changing a system's hardware means entirely redesigning the software in a new buzzword language. If they decide to revamp the current programs in the original language, they assume the cost will be such that they might as well have been rewritten the software from scratch.

In software manager George Holt's experience, porting Ada code to the latest architectures is not only reasonable technologically and financially, but can be done rapidly. This is true even when the latest technology is the humble PC loaded with the operating system and software that everyone uses. His team's work resulted in the first tactical command and control weapon being run on a PC.

Now president of AdaRose, Inc., Holt was a software manager for Mei Technology Corp. when the US Army asked that the Paladin Howitzer Artillery Vehicle be upgraded. In a matter of months his team put together a working prototype for PM Paladin at Picatinny Arsenal, NJ. They proved to the many skeptics among the Army's software managers that they could port the Ada83 code from a proprietary OS to Ada95 running on a PC's Pentium processor under Microsoft Windows NT.

The Paladin is an Army Artillery System mounted on a tracked vehicle (many civilians mistake it for a tank with a huge cannon). It fires 155mm (about 6.2 inch diameter) shells to a range of 30 kilometers. It has onboard navigational and automatic fire control systems, a four-man crew, and weighs approximately 62,000 lbs with a cruising range of 186 miles.

The Paladin can operate independently. From on the move, it can receive a fire mission, compute firing data, select and take up its firing position, automatically unlock and point its cannon, fire and move out--all with no external technical assistance. Paladin is capable of firing up to four rounds per minute. The Paladin features increased survivability characteristics such as day/night operability, nuclear, biological, and chemical protection with climate control and secure voice and digital communications.

The Paladin/FAASV is an ammunition resupply vehicle fielded with the Paladin that contains many artillery rounds and is capable of fast-feeding ammunition to the vehicle.

The display unit is menu driven with ASCII text. The operator uses no keyboard, but instead has a numerical keypad for entering numbers, and up and down and right and left arrow keys with which to step through the menus. Soft keys are situated below the display and their function changes according to the menu. The menu system allows entry of important info such as target coordinates and allows the operator to elevate the gun to a certain position to make it easier to load ammo into the cannon; he presses another key for aligning to the target. One-third of the way up the cannon's barrel sits an extremely accurate inertial navigation unit that tracks where the gun is in space (x,y,z coordinates). This positional data is fed to the software to align to the gun before firing. The Nav Unit uses a three-ring laser gyro and is produced by Honeywell.

The previous and current Ada software controls the Paladin's automatic firing control system functions, such as interactions with a number of hardware components; an inertial navigation unit with GPS that feeds into the software; command, control, and communications; digital and voice radio links to a command center; targeting info through digital means like the internet; commands to fire on a specific target; computing ballistic solutions; auto positioning the gun in elevation, and azimuth for aiming on the target.

All the major functions are contained in 225,000 lines of Ada, counted as carriage returns with no comments. The main kernel input/output (KIO) driver that acts

as interface between the hardware and software is written in C.

Originally in Ada83 with a significant amount of assembler language, the code interacted with three separate computers connected on a 1553 communication bus. The heart of the new system is an Intel-based automatic fire control computer that replaced the three line replaceable computer units while maintaining existing user interface, operational procedures, and tactical functionality. The conversion to Ada95 was "pretty straight forward though not automatic," according to Holt in a recent interview. "It involved a lot of manual smoothing and adding more functionality," which pumped up the project by an extra 25,000 Ada lines.

Luckily, some software engineers on the project had been experts at assembler and were proficient in Ada95. The assembler was hardware oriented, and those components were no longer operational, but the functionality had to be maintained. "You needed the speed then but today you don't," Holt explained. "Processors are powerful and cheap now and we really exploited their capabilities."

The port to the Pentium processor and NT OS created a unique situation, being the first time a tactical weapon system ran on a PC platform. The obstacles proved to be more from a lack of confidence, by many, in the power of the newest off-the-shelf architectures. Windows NT is obviously not a realtime OS (RTOS), which most weapons systems require, and therefore, the logic continued, the engineers would not be able to multi-task in real-time. Holt's team convinced the government that it did not need an RTOS because of the Pentium's speed, which for this particular weapon system multitasks in real-time.

The Mei Technology Corp engineers had to produce a proof of concept first to see if the modified software could run on NT. Within a few months after receiving the go-ahead, the engineers mounted the Paladin vehicle with the prototype software, hooked it up to the system, and ran a mission through a laptop with the converted code. After this proof-of-concept, the engineers, teamed with HW Vendors and Army Engineers from Picatinny Arsenal, developed and fielded the new system in less than two years.

Once awarded the contract, the team kept their approach to acquisitions, using commercial off-the-shelf technologies and open-system architecture designs, in accordance with DoD reform initiatives. As a result, the M109A6 Paladin now has a fire control computer with superior speed, expandability, and logistics supportability. All of these features were realized while saving \$27.5 million in production costs and reducing the computers' lifecycle costs by 75 percent. These savings were realized by leveraging

several key industrial-based technologies such as using a Pentium processor, the Ada programming language, and Windows NT, and combining them to meet the rigors of the military environment.

The engineers, many of whom now work for AdaRose in its Lexington, Mass., headquarters, played a major role in this revolutionary rehosting of an Army command and control system. The Paladin Automatic Fire Control System software was the first weapons system to be ported to an open architecture Pentium based computer running on Windows NT. The engineers' performance in creating this first was key in contributing to the Defense Standardization Program Outstanding Performance Award that was subsequently awarded for the milestone effort. Holt and the team's Technical Director, Ken Curry, accepted the award for Mei Technology.

Use of the Ada programming language and strong modular programming techniques allowed the team to rehost the legacy Ada software from the three-box distributed system to the new, one-box platform. Once coupled with the use of a Pentium processor and Windows NT, the team, in turn, exported the new software to similar U.S. military equipment, with minimal development costs.

Holt says that their work with Ada went well. Part of the software engineers' success came from using an Aonix Active Ada compiler and the ClearCase configuration management system, as well as their knowledge of Ada. While not everyone they hired was an Ada programmer, most had experience in Pascal or C. "Converting from another language to Ada is not as hard as teaching someone who has never programmed before," Holt said. "Just like a good mechanic can work on a Mercedes Benz or a Chevrolet, so a good programmer can work in Ada or C."

He says that programmers must be careful when using existing design documents. "Very often, in the military world your design documents may not be telling you what the code's doing. With legacy software any changes to the code may not have been reflected in the existing design documentation. However, a good programmer can look at Ada code and get an excellent feel for what the code is doing."

"To keep future engineers informed of the software changes, the Paladin team early in the project took the propriety software, analyzed each module, and redid all of the headers to provide more precise and up-to-date information. When programmers keep the headers up-to-date it provides a better understanding of the existing design documents, and a much easier path to bring these documents up to date. This

leads to good fielding and lifecycle maintenance."

It also leads to making the next port to future everyday architecture even easier.

(For more information, see AdaRose <<http://www.adarose.com/>> or contact Ann S. Brandon <webmaster@adaic.com>, Communications Director, ARA <<http://www.adaresource.com/>>)

Software Bugs Cost Big Bucks

From: Warren W. Gay VE3WWG
<ve3wwg@cogeco.ca>

Date: Mon, 09 Dec 2002 14:58:18 -0500

Subject: Software Bugs Cost Big Bucks
Newsgroups: comp.lang.ada

(Quoted from "CITO Linkline - November 29, 2002", see www.cito.ca)

"Software Bugs Cost Big Bucks

A recent study by the U.S. National Institute for Standards and Technology found that software glitches cost the American economy \$59.5 billion annually. The study suggests that better testing during software development could reduce that cost by a third, or \$22.2 billion per year. According to the study, 80 percent of the cost of developing software goes into discovering and fixing bugs. Because of this, there is an increasing interest in development tools and techniques to both reduce the cost of software development and foster the creation of more robust, reliable code."

Of course, the usual response to this is that "there is an increasing interest in development tools and techniques". But the Ada word never gets mentioned. Yet the words "to both reduce the cost of software development and foster the creation of more robust, reliable code" just seems to scream "Ada". *Sigh*

Notice particularly the statement "80 percent of the cost of developing software goes into discovering and fixing bugs". Again... I don't need to say it here.

Maybe with A#, at least the MicroSofties will become more aware of "other alternatives"? And maybe they'll tell 2 friends...

[And from another message: -- dc]

I had intended to leave a link to the full article:

<http://www.embedded.com/story/OEG20021127S0037>

Even this next paragraph screams "Ada", but only C/C++ gets mentioned in the article:

"Bugs flow downstream

The NIST study that came out in June found that more than half of all errors in software are not discovered early enough in the development process, but crop up "downstream," when the package is

nearing production, or even later, in the field. One way of making code writing more efficient is to integrate and test software components almost from the onset of an embedded-software development project. [...]"

On Compilers and Standards

From: Ted Dennison

<dennison@telepath.com>

Date: Sat, 04 Jan 2003 18:04:12 GMT

Subject: Re: Anybody in US using Ada?

Newsgroups: comp.lang.ada

[See also "Language Standards and Complexity of Compiler Writing" in AUJ 22-2 (Jun 2001), pp.94-95. -- dc]

> Within a week of starting to program in Ada [in 1988! -- dc] I found legal code that one or the other compiler rejected.

That's hardly anything you wouldn't have to deal with using C though. I believe there **was** a C standard in 88, but you wouldn't have known it to look at the compilers available. The same goes today for C++. The most commonly used C++ compiler today (MSVC++ 6.0) is just barely over 60% compliant, which means its barely more of a C++ compiler than not. I can tell you from sad experience that trying anything more than the most basic of the STL examples in Stroustrup's book will fail miserably with it.

> Most of the problems with C were solved by the introduction of C++.

Having worked with both, I'd say that is just true. However, 60% less of C's language-induced problems is not near enough. Plus, C++ adds several new ones of its own, and even makes some of C's existing problems far worse. (eg: Now unexpected implicit type casts can happen with **any** type, not just the numeric ones)

[About Ada not having C++'s automatic template function instantiation: -- dc]

MSVC++ 6 doesn't really have this feature either. A large amount of the compilers that do have it, do it differently from each other. That effectively means you can't count on it in portable code. That's why rule #1 in the Mozilla C++ portability guide is "Don't use C++ templates". (see

http://www.mozilla.org/hacking/portable-cpp.html#dont_use_templates)

[In another message, responding to the complaint that this document is almost five years old: -- dc]

...and yet is still linked to on the main programming page. That's because nothing in there has needed changing. That is **still** the policy at Mozilla.

> Today most template code is portable, except for a couple of features not implemented by MSVC 6.0.

More than a couple. None of the function generators work at all (try using a "for all" or "for each" template to insert something into or delete something from a container based on a predicate, which is pretty much what they were created for. It doesn't work.), and automatic instantiation doesn't work except in trivial cases. Large portions of boost are unusable too. Lots of the stuff that **is** usable requires you to manually specify the template parameters (because its auto-instantiation sucks). The only way you could possibly make any reasonably complicated template code "portable" is with conditional compilation, or by going out of your way to use the least-common denominator. With 2 or 3 platforms that you know about ahead of time you might be able to do that. Mozilla has to work with so many different sucky C++ compilers that the least common denominator is no templates at all.

[...] Bindings to anything that has a C interface are almost trivial to generate.

> Do you mean they can be generated automatically?

Well... yes there are such generators about, but no, that's not what I meant. I typically prefer to create my **own** bindings by hand, and have no compunction about doing so, even when good bindings are already available. That's what I mean by "almost trivial". Its about the same amount of work as writing external function prototypes in C.

On Language Perception

From: Bill Findlay

<BillFindlay@blueyonder.co.uk>

Date: Fri, 03 Jan 2003 04:43:29 GMT

Subject: Re: Anybody in US using Ada?

Newsgroups: comp.lang.ada

[Referring to a "language controversy" in the comp.arch newsgroup, it was noted that antipathy to Ada was very often irrational. -- dc]

Opposition to Ada in comp.arch fell into three categories, which I parody (grotesquely unfairly, I admit 8-) as follows:

(1) "We don't care about software quality. We make money selling **** written in C, and that's fine with us."

(2) "We do care about software quality. We write our software in C (or other, even less safe, languages) and ensure its quality by being faultless programmers and superior human beings. Ada is for talentless losers."

(3) "Ada is too low-level. Our favourite language is Functional-Telepathy/1, which generates an optimal program for you while you are still thinking about the specification. It should be implemented real soon now."

Depressingly, type (1) critics were in a majority. When shown evidence (the

Rational data) that Ada could help them to make even more money, the response was "I don't believe it"; taken even to the point of suggesting that Rational had fabricated their figures. In other words: "I've no evidence of my own, so I'll find reasons to ignore yours". The intensity of denial was astonishing.

Type (2) and type (3) critics tended to post from academic domains [...], although academia did not have a monopoly on false pride.

It's interesting that essentially no-one objected to Ada on the grounds of technical or pragmatic issues such as are openly discussed here.

> I don't think Ada is **bad** -

That's damming with faint praise. I think Ada 95 is very, very good indeed.

> but it sure has a problem selling itself with "reliability", etc.

[...] I want the general level of software quality to rise, and I believe that better understanding and wider adoption of Ada would promote this objective. I'm more than happy to make common cause with anyone [...] who has ideas about how to make that happen. But I do not think that it is Ada that is the barrier; [...]

From: Bill Findlay

<BillFindlay@blueyonder.co.uk>

Date: Fri, 03 Jan 2003 18:28:56 GMT

Subject: Re: Anybody in US using Ada?

Newsgroups: comp.lang.ada

> [...] I've had my own data showing how Ada can make more money and the problem is that this sort of data just doesn't seem to help people get interested in it. Mostly because it is "Life Cycle" money and most business efforts are driven by "Time To Market" money. They don't care if it costs more in the long run because if it doesn't get there first, it won't earn anything anyway, so better to spend more "in the long run" if whatever you do gets a product out the door quicker.

That pretty much sums up what type (1) critics said. Despite my making fun of them, I do think they have the makings of a point. But don't you think that (other things being equal - and that is what you are emphasizing) Ada **would** get them to market sooner?

I was even more depressed that type (2) critics, who did care about quality, had the perception that Ada was a crutch for the talentless. This is so contrary to my own perception that I find it bewildering.

Choosing the Language for a Project - Tales from the Trenches

From: Jerry Petrey

<jdpetrey@raytheon.com>

Date: Tue, 07 Jan 2003 13:34:59 -0700

Organization: Raytheon Company
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

> How is such a decision (choosing the language for a major project) made, especially when the language chosen has not been proven in such applications? [...]

Having worked for Lockheed and a number of other defense companies in the past, I can comment on [this] question.

Usually the managers who make the decisions on such things as languages, tools, processors, etc. don't have a clue what is the best choice. They are not particularly interested in things like reliability, maintainability, safety, or quality; rather, what they think will be fast and cheap and make them look good in the short term (the old "You guys start coding, while I go upstairs and see what they want us to build"). They tend to listen to the engineers they perceive as being knowledgeable (in many cases these are relatively young engineers who have done a good job of kissing up and winning the favor of the managers). These engineers, in turn, are thinking of what will be the best for their career and what they are most familiar with. These are the guys we have to convince that Ada is the right choice, but it is very difficult to do since they are often ignorant of Ada and firmly believe that due to the popularity of C++, it is the right choice for new applications.

Many previous posts have suggested ideas for improving Ada's acceptance - cheaper and better compilers and tools, bindings, etc. These are all good, but most of all we need to get Ada more into the education system and promote the successes of Ada applications that are done. And, of course, do more Ada applications where we have the control to do so - at work or at home. Having Ada available on more commonly used platforms would certainly be a big step as well.

Jerry Petrey, Senior Principal Systems Engineer,

Raytheon Missile Systems

From: Dennis Lee Bieber
<wlfraed@ix.netcom.com>

Date: Tue, 07 Jan 2003 16:39:38 -0800
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

Sounds familiar... In the mid-80's I was in a program that had ~100 programmers: 80 Fortran coders of various skill levels (I'm sorry to say we had a few, fortunately short-lived, who /still/ didn't understand just what call-by-reference meant, and one who built a 35-entry if/elseif/endif structure (and 35 subroutines!) to handle data that was a 7-slot array of records with 5 variants), and 20 PDP-11 assembly programmers. The assembly did realtime code.

A study was done to determine the language to be used for an upgrade from PDP-11 to microVAX. The candidates: Fortran, Assembly, Pascal, and C. Intelligently, they rejected C as too error-prone, and Assembly as too expensive (nobody skilled in VAX-11 assembly). Amazingly, they also rejected Fortran (80 skilled programmers available to assist the transition) as not modern. They chose Pascal as modern, and they could get new-hires who had learned the language in college.

I had two responses to that: 1) you expect new hires who know (for the time period) Turbo-Pascal to code realtime logic on a VAX; 2) if you were going to go the mile for Pascal, why not fall over the extra foot to pick up a language the fixes Pascal's flaws /and/ was designed for realtime -- Ada.

Hearsay is the manager who finalized the decision for Pascal recanted the decision a decade later -- when it was too late to get rid of the remaining Pascal (and when much of the system underwent a third architecture change to HPUNIX/C).

The Money Argument

From: John R. Strohm

<strohm@airmail.net>

Date: Sat, 18 Jan 2003 15:39:12 -0600

Subject: Re: advantages or disadvantages of Ada over Pascal or Modula
Newsgroups: comp.lang.ada

> Assume you have to program a brake control for car, would you do it in Ada? [...] C systems are cheap and even free available. Modula systems are expensive. But - Ada prices are far beyond from good and bad. Have you ever heard the prices from [...]? Embedded systems in non military environment have not some k-dollars available for sw-tools.

Have you ever been exposed to basic engineering economics? In the United States, with all overhead considered, a good engineer costs his employer on the rough order of \$250,000 per year. I would be very surprised if the price was that much different in Germany. When you are talking about spending that kind of money on software engineers, trying to pinch pennies on their tools is idiocy. What you want is tools that will improve their productivity, that will let them get more work done per year.

Recall that Pratt & Whitney documented 2x productivity improvement using Ada. 2x productivity improvement, when an engineer costs \$250,000/year, means that an Ada toolset that costs \$250,000 per engineer PAYS FOR ITSELF IN ONE YEAR. If two engineers use that toolset, and they both get 2x productivity improvement, the toolset PAYS FOR ITSELF IN SIX MONTHS. With ten engineers using the toolset, and only a

10% improvement in productivity, that \$250,000 toolset STILL pays for itself in a year.

This, incidentally, is a significant part of the reason why Symbolics LISP machines, single-user workstations with high five-digit pricetags, sold faster than beer at Wurstfest parties. They paid for themselves in months, at a time when fully-burdened engineers cost about \$100,000 per year.

AND FURTHERMORE.

When you are talking specifically about brake controls for cars, you have to recognize a few things. First, you are talking about a safety-critical system. Anything less than the absolute best NO-FAIL quality is unacceptable, because people will get killed AND THEIR SURVIVING RELATIVES WILL SUE YOUR COMPANY OUT OF EXISTENCE. Worse, if they can find the registered professional engineer who signed off on the brake control software, HE WILL GO TO PRISON for involuntary manslaughter. The accepted standard in the United States, last I heard, was around \$70,000 per injury and \$300,000 per death. Second, you are going to amortize your development costs over hundreds of thousands of cars, so a toolset that costs \$100,000 winds up costing PENNIES per car. Third, when those cars start rolling off the assembly line, if your brake controls are not ready, it will cost the car manufacturer a small fortune EVERY DAY your controls are late.

In other words, trying to save a few pennies on software development toolsets is sheer idiocy.

From: John R. Strohm

<strohm@airmail.net>

Date: Sat, 18 Jan 2003 17:16:18 -0600

Subject: Re: advantages or disadvantages of Ada over Pascal or Modula
Newsgroups: comp.lang.ada

> ACK. But you forgot the usual manager thinking.

No, I didn't. Instead, I explain the usual manager thinking, and the way the software engineer must couch the argument so as to work WITHIN the usual manager thinking.

Let me give you an example. A buddy of mine once did a contract gig at GM-Delco, several years ago. The project was embedded software for a heads-down display for the new Cadillac.

He noticed something interesting. When an engineer told his manager that he needed a certain piece of test equipment, it showed up in the lab, the next day, powered up and ready to go. When he said he needed prototype hardware for debugging, a wire-wrapped prototype showed up in a couple of days, and a

printed circuit board showed up a few days later.

He inquired, and management explained. In eighteen months, new Cadillacs were going to come rolling off the assembly line. That software HAD to be there, or those cars couldn't be sold. Every day those cars couldn't be sold, General Motors would incur VERY, VERY LARGE interest expenses. GM-Delco had been told that they would NOT be the cause of those expenses.

In short, make a GOOD business case that Ada will save time and money, quantify the savings, and point out the costs of missing the deadline.

Or you could put it this way: "If we do this in C, we're going to have higher defect rates. If those defects kill someone, there will be an inquiry. Do you really want to be identified, in court, as the man who caused the higher defect rate, that generated the defect, that killed that young mother? Or would you rather be the manager who can honestly sit in that witness stand and say he did EVERYTHING he could to ensure that the defect rate would be as low as possible?"

From: Marin David Condic
<mcondic@acm.org>

Date: Sun, 19 Jan 2003 11:31:21 -0500
Subject: Re: advantages or disadvantages of Ada over Pascal or Modula
Newsgroups: comp.lang.ada

Just remember that Ada tends to show its cost and defect advantages in large scale projects rather than small ones. On large scale projects, there are also lots of other factors (such as process, design, etc.) that can be far more important to reliability & schedule - Ada can't cure that either.

If the tools are available, the Ada case gets easier, but it still has to overcome institutional inertia and lack of software engineering culture that may exist in many shops.

From: John R. Strohm
<strohm@airmail.net>

Date: Sat, 18 Jan 2003 23:51:23 -0600
Subject: Re: advantages or disadvantages of Ada over Pascal or Modula
Newsgroups: comp.lang.ada

You have to learn [the business world's] language. Let me give you an example. A whole bunch of companies spent a lot of money on Expert Systems work in the 1980s because of a couple of huge success stories. Digital Equipment Corporation did the VAX configurator, that saved them a huge boatload of money. BBN did a geology expert system that found a monster ore deposit that humans had overlooked. When you show a beancounter that you can make him REAL MONEY using technology, he'll listen.

From: Marin David Condic
<mcondic@acm.org>

Date: Sun, 19 Jan 2003 11:43:29 -0500
Subject: Re: advantages or disadvantages of Ada over Pascal or Modula
Newsgroups: comp.lang.ada

Absolutely! Yeah, verrily! And Amen! Engineers tend to think that the object of the game is to use really advanced tools to build really cool stuff. Bzzzzttt! Wrong Answer! The object of the game is to MAKE MONEY and the sooner Ada can start showing people how it MAKES MONEY, the sooner it will start finding acceptance against all of the objections of the anti-Ada software developers that don't want to use it. If the industry that manufactures blivets uses predominantly C and someone starts a business to make blivets and uses Ada and starts coming out with better, less expensive blivets that show up earlier to market, you'll see Ada catch on in the blivet industry like wildfire. [...]

From: John R. Strohm
<strohm@airmail.net>

Date: Sun, 19 Jan 2003 17:26:01 -0600
Subject: Re: advantages or disadvantages of Ada over Pascal or Modula
Newsgroups: comp.lang.ada

That was PRECISELY what Rational did with the R1000, back in the mid-1980s.

I really wish I had taken the time to research the business case, and do a proper chart pitch on it, with supporting data in detail. I sketched it out on the back of some piece of paper or another, but I never got around to fleshing it out.

The numbers Rational was disseminating, which were backed up by other organizations who gave the product an honest try, were just-plain phenomenal. As in "you don't have to build a new building to house the next project" phenomenal. I think I could have made the case JUST on parking lot costs at GD/FW.

The Training Argument

From: John R. Strohm
<strohm@airmail.net>

Date: Fri, 3 Jan 2003 04:11:08 -0600
Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

> Regardless of any superiority in reliability, the scarcity of [...] trained programmers makes Ada too expensive [...]

In the very early 1980s, General Dynamics / Fort Worth Division started the F-16C/D program. This was a MAJOR upgrade of the airplane, involving, among other things, all new computers and all new software.

Ada wasn't there yet, so they chose JOVIAL J73.

At that time, there existed precisely one J73 compiler, and it didn't target EITHER of the processors they were designing into the airplane (Zilog Z8002 and MIL-STD-

1750A). They wound up having to let compiler development contracts to two (small) companies to develop toolsets.

At that time, trained J73 programmers just plain didn't exist. GD/FW had to train every single programmer they hired for that project.

Every time I hear someone grumbling about the scarcity of trained Ada programmers, I think about F-16C/D and JOVIAL, and I wonder how GD/FW ever managed to get that airplane off the ground, if training is so hard.

I'm not even going to mention HAL/S, the Space Shuttle language, which to my (unclassified) knowledge was not used for anything else on the planet. Where does NASA (and the SEI Level 5 contractor) get trained HAL/S programmers?

A Few Quotes

From: Ted Dennison
<dennison@telepath.com>

Date: Wed, 08 Jan 2003 13:14:47 GMT
Subject: Re: Is Ada a good choice for a beginner to programming?
Newsgroups: comp.lang.ada

> [...], if you are a hobbyist, an academic or a government contractor, well you can ignore the market place.

If you are writing something that has to be around a while, it in fact would be *wise* to ignore the marketplace. If all your tools have going for them is fashion, they will disappear when the direction of the wind changes.

From: Jeffrey Carter <jrcarter@acm.org>
Date: Fri, 17 Jan 2003 15:31:09 -0700
Subject: Re: Ada -- Increase or Decrease to Risk?

To: team-ada@acm.org

> [Upon proposing Ada, I was] asked for justification for using what was thought to be unproven technology.

What technology do they consider "proven"? [...] C is proven technology. Proven to increase errors and costs, compared to Ada.

Choose Ada

From: Richard Riehle
<richard@adaworks.com>
Date: Tue, 07 Jan 2003 04:38:19 -0800
Organization: AdaWorks Software Engineering

Subject: Re: Anybody in US using Ada?
Newsgroups: comp.lang.ada

Programmers, most of them I know, don't care a whit about reliability. They care about convenience. Remember when we would teach Ada 83 and we had to show the students how to instantiate a generic package so they could do simple I/O? Remember how difficult it was, with Ada 83 to do simple file management? Remember how frustrating it was to

format an MS-DOS screen using standard Ada? Remember how the students reacted when exposed to the visibility rules?

Most of those new Ada programmers just wanted to write simple programs to get started. Many wanted to use Ada on their home machines to do little projects for fun. I recall the Meridian compiler had a DOS package that was pretty good, but still fell short of what we needed. For example, with the overpriced Alsys compiler, one could access any part of memory because the type Address allowed it. With the Meridian compiler the type Address was a type signed integer so one could not directly address the display address values in the upper 360KB of DOS space. Sounds silly. Maybe, but it prevented one of my customers from doing the kind of high-performance graphics they could do with Alsys.

We have defined the language for technical people and made it almost inaccessible to those who simply want to do the kind of programming they usually do with Fortran. We have some libraries, finally, for doing GUI programming, but too late to inspire the ordinary programmer to take a second look. My students are at different levels of capability. Some love to program with John English's JEWEL package since it so easy to do so many interesting Windows programs. Others love CLAW. At least one pair of students did a great M.S. thesis using GtkAda.

One of the saddest things was the abandonment of the Java Byte Code initiative from GNAT. Yes, I realize there was insufficient commercial demand. However, I had generated a lot of interest in it at NPS until it turned out that the existing version was not up-to-date with the current JVM capabilities. My colleagues, at first enthusiastic, quickly lost interest. In the parlance of the "romance novel" this is known as "seduced and abandoned."

Ada is, in my view, the best choice of programming languages for a wide range of applications, wider than reliability-oriented applications, and wider than most other languages. I was just reading my new copy of the Consolidated Ada Language Reference Manual at 2:00 AM this morning and found myself entertained by the improvement in language, explanation, and clarification. We need more literature explaining, in depth and with examples, how certain things work. For example, someone could do a good tutorial on the mathematical capabilities of the language using some examples from the Numerical Recipes Series. I tried to create a little booklet with my Ada Distilled to make the language more accessible. The responses I get from readers of Ada Distilled has been encouraging and I continue to update and expand it based on that feedback.

We need more libraries similar to those created by Jerry Van Dijk, such as his NT_Console package. In fact, that package needs a little enhancement, but is pretty good as it is.

I have mentioned before, and re-emphasize again, those of us who want to promote Ada can do nothing better than create commercial applications using it. There is not end to the ideas for new applications and no end to what can be sold in the marketplace. Where are the games programmed in Ada and sold in shrink-wrap? Where are the database applications created in Ada and marketed to the business community? Where is the spreadsheet program written in Ada that makes Excel look like yesterday's stale bread?

The best way to sell Ada is to use it for building applications that people use on their computers. CLAW is a great starting place for creating such applications for Windows. GtkAda is a great starting place to create the sorely needed applications for Linux, a platform where users are desperate for new applications.

The path to commercial success for Ada is to use it for successful commercial products. Harness your own creativity to Ada. Build products. Don't tout them as Ada products but as products that solve a problem. Only after your product is wildly successful do you need to reveal that you used Ada.

It is nearly impossible to convince a large corporation to change course in its choice of programming languages once the bureaucracy has made its decision. Such decisions are rarely made on technical grounds. They are usually made by people who have little knowledge of software and less knowledge of programming languages. They make incredibly stupid decisions, most often choosing something as disastrous as C++. I often suggest to my students that, if C++ is the best we can do in programming languages, this industry is in trouble for a long time to come.

Ruby is achieving success because developers are choosing it. Python is successful for the same reason. Eiffel, which should be more successful, is not as widely received, in part, I suspect because it is viewed as an expensive commercial alternative. Ada is now available in free, downloadable versions similar to Ruby, Smalltalk, Scheme, and Python. That is a good start, but not quite enough.

Go and make your fortune with a product created in Ada and you can walk away from that dour, windowless cubicle in which you have been imprisoned by some huge defense contractor. You can build as good or better products using what we now have available in the Ada language and its available libraries.

Programmers of the world, choose Ada. You have nothing to lose but your chains.

Conference Calendar

This is a list of European and large, worldwide events that may be of interest to the Ada community. More information on items marked ♦ is available in the Forthcoming Events section of the Journal. Items in larger font denote events with specific Ada focus. Items marked with ☺ denote events with close relation to Ada.

The information in this section is extracted from the on-line *Conference announcements for the international Ada community* at: <http://www.cs.kuleuven.ac.be/~dirk/ada-belgium/events/list.html> on the Ada-Belgium Web site. These pages contain full announcements, calls for papers, calls for participation, programmes, URLs, etc. and are updated regularly.

2003

- April 02-04 4h **International Conference on Software Testing (ICSTEST'2003)** Cologne, Germany
- April 05-13 **European Joint Conferences on Theory and Practice of Software (ETAPS'2003)** Warsaw, Poland. Event includes: conferences from 7 to 11 April 2003, affiliated workshops on 5-6 and 12-13 April, 2003. Includes a.o. the following events:
- April 06 ETAPS2003 - 3rd **Workshop on Language Descriptions, Tools and Applications (LDTA'2003)**. Topics include: Program analysis, transformation, generation; Automatic generation of language processing tools; etc.
- April 06 ETAPS2003 - **Workshop on Software Composition (SC'2003)**
- April 07-11 ETAPS2003 - 12th **International Conference on Compiler Construction (CC'2003)**. Topics include: compilation and interpretation techniques; run-time issues; language constructs and their implementation; modularization constructs and techniques for separate compilation; tools for compiler construction or language support, including debuggers, profilers, refactoring tools, etc.
- April 07-11 ETAPS2003 - 12th **European Symposium on Programming (ESOP'2003)**. Topics include: design of programming languages and calculi; techniques, methods and tools for their implementation; exploitation of programming styles within different programming paradigms; automatic and manual methods for reasoning about programs; etc.
- April 07-11 ETAPS2003 - **Fundamental Approaches to Software Engineering (FASE'2003)**. Topics include: experience reports on best practices with component models and specifications, development tools, modelling environments, and software development kits; etc.
- April 07-11 ETAPS2003 - 9th **International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'2003)**
- April 07-11 10th **IEEE Symposium and Workshops on the Engineering of Computer Based Systems (ECBS'2003)** Huntsville, Alabama, USA. Topics include: Component-Based Design and Reuse; Middleware for Embedded Systems; Applied Formal Methods; Education and Training; Embedded Systems; Evolution, Reengineering and Legacy Systems; Reliability, Dependability, Safety and Security; Verification and Validation Standards; etc.
- April 08-10 7th **International Conference on Evaluation and Assessment in Software Engineering (EASE'2003)** Staffordshire, UK. Deadline for submissions: April 2, 2003 (posters)
- ♦ April 09-10 **Spring Ada UK User Group conference**, Swindon, UK. Conference on April 9th, post-conference workshop on Ada 0Y on April 10th (led by Tucker Taft)
- April 14-16 10th Annual **European Concurrent Engineering Conference (ECEC'2003)** Plymouth, UK
- ☺ April 22-26 **International Parallel and Distributed Processing Symposium (IPDPS'2003)** Nice, France. Topics include: Parallel and distributed software, including parallel programming languages and compilers, operating systems, runtime, middleware, libraries, programming environments and tools for parallel and distributed computing, etc. Includes a.o. the following events:

- April 22-26 IPDPS2003 - 8th **International Workshop on Formal Methods for Parallel Programming: Theory and Applications** (FMPPTA'2003)
- April 22-26 IPDPS2003 - 4th **Workshop on Parallel and Distributed Scientific and Engineering Computing with Applications** (PDSECA'2003)
- ☉ May 03-10 25th **International Conference on Software Engineering** (ICSE'2003) Portland, Oregon, USA. Includes a.o. the following events:
- May 03 ICSE2003 - **Workshop on Software Architectures for Dependable Systems** (WADS'2003)
- ☉ May 09-10 ICSE2003 - 3rd **International Workshop on Software Engineering for High Assurance Systems** (SEHAS'2003). Topics include: specification, validation, verification, testing, and certification of high assurance systems; the role of tools in support of high assurance system development; convincing case studies that apply systematic methods to the construction of high assurance systems; etc.
- ☉ May 14-16 6th IEEE **International Symposium on Object-oriented Real-time distributed Computing** (ISORC'2003) Hakodate, Hokkaido, Japan.
- May 19-22 23rd **International Conference on Distributed Computing Systems** (ICDCS'2003) Providence, Rhode Island, USA. Topics include: Middleware; Fault Tolerant and Dependable Systems; Real-time and Embedded Systems; Software Engineering and Formal Methods; etc.
- May 26-29 4th **International Conference on eXtreme Programming and Agile Processes in Software Engineering** (XP'2003) Genoa, Italy
- ☉ May 27-30 9th IEEE **Real-Time and Embedded Technology and Applications Symposium** (RTAS'2003) Toronto, Canada. Topics include: Real-time Linux and applications; DRE middleware, e.g. Real-time CORBA; Secure real-time systems; Real-time software component models; QoS-aware application design and patterns; Embedded control applications; etc.
- May 28-30 7th **Brazilian Symposium on Programming Languages** (SBLP'2003) Ouro Preto, MG, Brazil. Topics include: Programming language design and implementation; Formal semantics of programming languages; Programming languages for mobile, WWW, and network computing; Teaching programming languages; etc.
- June 08 2nd **International Workshop on Distributed Event-Based Systems** (DEBS'03) San Diego, California, USA
- June 09-13 **International Conference on Practical Software Quality Techniques & Testing Techniques** (PSQT/PSTT'2003 East) Washington, DC, USA
- June 09-14 28th Annual **USENIX Technical Conference** (USENIX'2003) San Antonio, Texas, USA. Topics include: applications, architecture, implementation, and performance of modern computing systems; Reliability and QoS; Interoperability of heterogeneous systems; special FREENIX track on freely redistributed technology; etc.
- June 11-13 10th **International Static Analysis Symposium** (SAS'2003) San Diego, California, USA. Topics include: abstract interpretation, data flow analysis, abstract testing, security analysis, distributed systems, embedded systems, etc.
- ◆ June 16-20 8th **International Conference on Reliable Software Technologies - Ada-Europe'2003**, Toulouse, France. Sponsored by Ada-Europe, in cooperation with ACM SIGAda. Topics include: Management of Software Development and Maintenance; Software Quality; Software Development Methods and Techniques; Software Architectures; Tools; Kinds of Systems; Applications; Ada Language and Tools; Ada Experience Reports; Education and Training; Case Studies and Experiments; and a special session on Avionics and Space, including the use of Ada in this realm. Includes a.o. the following event:
- June 20 Ada-Europe'2003 - **Workshop: Quality of Service in Component-Based Software Engineering** (QoSCBSE'2003)

- June 16-17 **9th International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'2003)**. Klagenfurt/Velden, Austria. In connection with CAiSE'03.
- June 16-20 **ACM/IFIP International Middleware Conference (Middleware'2003)** Rio de Janeiro, Brazil. Theme: "Information Systems for a Connected Society". Topics include: Distributed real-time and embedded (DRE) middleware platforms; Reliable and fault-tolerant middleware platforms; Formal Methods applied to middleware; Novel paradigms, APIs, and languages for distributed systems; etc.
- June 18-20 **International Conference on Application of Concurrency to System Design (ACSD'2003)** Guimaraes, Portugal. Topics include: Real-time and hybrid systems; Case studies of concurrent systems design and verification; Presentation of software tools supporting the above topics; etc.
- June 22-25 **International Conference on Dependable Systems and Networks (DSN'2003)** San Francisco, CA, USA. Topics include: Architectures for Dependable Computer Systems; Fault Tolerance in Transaction Processing; Fault Tolerance in Distributed & Real-Time Systems; Safety-Critical Systems; Software Testing, Validation, and Verification; Software Reliability; etc. Deadline for submissions: April 21, 2003 (student forum, fast abstracts)
- ☉ June 23-26 **International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'2003)** Las Vegas, Nevada, USA. Topics include: Parallel/Distributed applications; Reliability & fault-tolerance; Real-time & embedded systems; Object Oriented Technology & related issues; Software tools & environments for parallel & distributed platforms; Education: parallel & distributed processing in computer science curriculum; Recent history (last decade) of parallel/distributed processing & what to expect in the next decade: New Horizons; etc. Session on High Performance Computing in Industry includes: Parallel Programming Languages; Software Tools and Programming Environments for Parallel/Distributed Platforms; Parallelizing Compilers; Operating System and Runtime Support for Parallel Computing; etc.
- July 13-16 **22nd Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing (PODC'2003)** Boston, Massachusetts, USA. Topics include: design, specification, implementation, application and theory of distributed systems
- ☉ July 14-17 **OMG Annual Workshop on Real-Time and Embedded Distributed Object Computing**, Washington, DC, USA. Topics include: Applying CORBA in any real-time or embedded system; Integrating CORBA with legacy real-time and embedded systems; Evaluation of real-time, high confidence, and embedded middleware; Advanced scheduling techniques and high-level real-time programming models; Fault-tolerance issues in real-time and embedded systems; etc.
- ☉ July 21-25 **17th European Conference on Object-Oriented Programming (ECOOP'2003)** Darmstadt, Germany. Includes a.o. the following events:
- ☉ July 21-25 ECOOP2003 - **Workshop on Exception Handling in Object Oriented Systems (EHOOS'2003)**. Topics include: formalisation, distributed and concurrent systems, practical experience, design patterns and frameworks, practical languages (Java, Ada 95, Smalltalk, Beta), etc. Deadline for position paper submissions: April 25, 2003
 - ☉ July 21 ECOOP2003 - **7th Workshop on Pedagogies and Tools for Learning Object-Oriented Concepts**. Topics include: frameworks/toolkits/libraries for learning support; approaches and tools for teaching design early; experiences with innovative CS1 curricula; etc. Deadline for position paper submissions: May 1, 2003
 - ☉ July 21 ECOOP2003 - **13th PhD Students Workshop in Object-Oriented Systems (PhDOOS'2003)**. Topics include: Concurrent, real-time, parallel systems; Patterns; Distributed and mobile object systems; Language design and implementation; Programming environments; Software components; etc. Deadline for position paper submissions: April 25, 2003
 - July 21 ECOOP2003 - **8th International Workshop on Component-Oriented Programming (WCOP'2003)**. Deadline for position paper submissions: April 14, 2003

- July 21 ECOOP2003 - 4th **International Workshop on Object-Oriented Re-engineering** (WOOR'2003). Deadline for paper submissions: April 14, 2003
- ☉ July 22 ECOOP2003 - **Workshop on Communication Abstractions for Distributed Systems** (CADS'2003). Topics include: Communication abstraction in programming languages; Middleware services; Design Patterns for communication and distribution; Communication components; Authentication, authorization, privacy; Group-oriented communication; Tolerance of Partial Failures; etc. Deadline for position paper submissions: April 25, 2003
- ☉ August 13-15 16th **International Conference on Parallel and Distributed Computing Systems** (PDCS'2003) Reno, Nevada, USA. Topics include: all areas of Parallel and Distributed Computing Systems, their modeling and simulation, design, use and performance, etc.
- ☉ August 24-27 6th **Joint Modular Languages Conference** (JMLC'2003) Klagenfurt, Austria. Topics include: concepts of well structured modular software; teaching good design and programming style; construction of large and distributed software systems; etc. Includes on 25-26 August, some common tutorials, keynote talks and a common panel with EuroPar'2003. Deadline for early registration: July 15, 2003
- ☉ August 26-29 9th **International Conference on Parallel and Distributed Computing** (Euro-Par'2003) Klagenfurt, Austria. Topics include: all aspects of parallel and distributed computing. Includes on 25-26 August, some common tutorials, keynote talks and a common panel with JMLC'2003. Deadline for early registration: July 15, 2003
- ☉ August 27-29 4th **International Conference on Parallel and Distributed Computing, Applications, and Techniques** (PDCAT'2003) Chengdu, PR China. Topics include: all areas of parallel and distributed computing, including Parallelizing compilers, Component-based and OO Technology, Programming languages and software tools, etc.
- September 01-05 Joint 9th **European Software Engineering Conference** (ESEC) and 11th **ACM SIGSOFT International Symposium on the Foundations of Software Engineering** (FSE-11) Helsinki, Finland. Includes a.o. the following event:
- September 01-02 ESEC/FSE-11 - **International Workshop on Principles of Software Evolution** (IWPSE'2003). Topics include: methodology for evolutionary design and development, validation and verification of evolution, experience and lessons learned from evolutionary software systems, etc. Deadline for submissions: May 18, 2003
- September 02-05 9th **International Conference on Object-Oriented Information Systems** (OOIS'2003) Geneva, Switzerland
- September 03-05 29th **EUROMICRO Conference** (EUROMICRO'2003) Antalya, Turkey. Includes track on: Component-based Software Engineering (Component design, implementation, testing; Development environment and tools; Case studies and experience reports; Components for real-time and embedded systems; etc.)
- September 08-12 11th IEEE **International Requirements Engineering Conference** (RE'03) Monterey Bay, California, USA. Deadline for submissions: April 26, 2003 (posters, research demos)
- September 08-14 12th **International Formal Methods Europe Symposium** (FME'2003) Pisa, Italy. Topics include: concerns and risks for potential adopters of formal methods; cost-benefit analysis; reports on practical use and case studies (reporting positive or negative experiences); tool support and software engineering; etc.
- ◆ September 15-19 12th **International Real-Time Ada Workshop** (IRTAW-2003) Viana do Castelo, Portugal. Deadline for position paper submissions: June 1, 2003
- September 16-19 7th **International IEEE Enterprise Distributed Object Computing Conference** (EDOC'2003) Brisbane, Australia. Topics include: Use and enhancement of middleware platforms; Practical experiences with enterprise distributed object computing; etc. Deadline for submissions: April 18, 2003 (tutorials)

- September 22-25 2nd **International Conference on Generative Programming and Component Engineering** (GPCE'2003) Erfurt, Germany. In Cooperation with ACM SIGPLAN and SIGSOFT. Deadline for submissions: April 6, 2003 (papers, workshops), May 18, 2003 (practitioner report, tutorials, demonstrations), July 13, 2003 (posters)
- September 22-26 IEEE **International Conference on Software Maintenance** (ICSM'2003) Amsterdam, The Netherlands. Includes a.o. the following event:
- ☉ September 23 ICSM2003 - **International Workshop on Evolution of Large-scale Industrial Software Applications** (ELISA). Topics include: empirical studies of evolving software, scalability of technological solutions, evolution of open source and COTS software, etc. Deadline for submissions: May 24, 2003
- ☉ September 23-26 22nd **International Conference on Computer Safety, Reliability and Security** (Safecomp'2003) Edinburgh, Scotland, United Kingdom. Topics include: state-of-the-art, experience and new trends in the areas of computer safety, reliability and security regarding dependable applications of computer systems.
- September 25-26 3rd **International Conference on Quality Software** (QSIC'2003) Beijing, China. Topics include: Debugging; Economics of software quality and testing; Formal methods; Quality evaluation of software products and components; Reliability; Software quality education; Static and dynamic analysis; Testability; Testing of object-oriented systems; Testing of concurrent and real-time systems; Testing strategies, tools, processes, and standards; Tool support for improving software quality; Validation and verification; Application areas such as component-based systems, distributed systems, embedded systems, enterprise applications, ...; etc.
- ☉ September 27-Oct. 01 12th **International Conference on Parallel Architectures and Compilation Techniques** (PACT'2003) New Orleans, LA, USA. Topics include: Programming languages for parallel scientific and object-oriented applications, etc. Deadline for submissions: April 4, 2003 (abstracts), April 11, 2003 (papers)
- ☉ September 29-Oct. 02 **GI-Jahrestagung Informatik 2003 - Teiltagung "Sicherheit - Schutz und Zuverlässigkeit"** Frankfurt/Main, Germany. Topics include (in German): Software in sicherheitskritischen Anwendungen, Zuverlässigkeit und Sicherheit softwarebasierter Systeme, etc.
- September 30-Oct. 01 IEEE **International Symposium on Empirical Software Engineering** (ISESE'2003) Frascati, Italy. Topics include: strengths and weaknesses of technology in use and new technologies, etc.
- ☉ October 01-03 9th IEEE Computer Society's **International Workshop on Object-oriented Real-Time Dependable Systems** (WORDS'2003) Capri Island, Italy. Deadline for submissions: April 10, 2003 (papers), April 20, 2003 (panels)
- ☉ October 06-08 22nd **Symposium on Reliable Distributed Systems** (SRDS'2003) Florence, Italy. Topics include: Distributed systems with properties such as reliability, availability, security, safety, and/or real time; Security and high-confidence systems; Analytical or experimental evaluations of dependable distributed systems; etc.
- October 06-09 13th **International Conference on Software Quality** (ICSQ'2003) Dallas, Texas, USA
- October 20-24 6th **International Conference on UML - Modeling Languages and Applications** (UML'2003) San Francisco, USA. Topics include: tool support for any aspect of modeling or model use; models in the development and maintenance process; domain-specific and concern-oriented modeling; etc. Deadline for submissions: April 8, 2003 (abstracts), April 15, 2003 (papers), April 21, 2003 (workshops), June 29, 2003 (tutorials)
- ☉ October 26-30 18th Annual **ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications** (OOPSLA'2003) Anaheim, California, USA. Deadline for submissions: April 13, 2003 (Domain-Driven Development special track papers), July 1, 2003 (Posters, Demonstrations, Doctoral Symposium, and Student Volunteers)
- ☉ October 27-31 22nd **Digital Avionics Systems Conference** (DASC'2003) Indianapolis, Indiana, USA. Software Engineering Track Topics include: Formal Specification and Verification; Software Reliability - Measurement and Techniques; Software Safety; Software Architecture - Evolution/Product Families; etc.

- October 28-31 **IEEE Symposia on Human-Centric Computing Languages and Environments (HCC'03)** Auckland, New Zealand. Topics include: tools that enable humans, using textual languages, visual languages or any other appropriate technologies, to accomplish their tasks more effectively; etc.
- ☉ October 30-Nov. 01 **International Conference on Compilers, Architectures and Synthesis for Embedded Systems (CASES'2003)** San Jose, California, USA. Topics include: Compilation techniques that focus on embedded architectures; Design, specification, and analysis of embedded systems; Managed runtime environments for embedded systems; Memory management and compiler controlled memories; Software design for multiprocessor embedded systems; etc. Deadline for submissions: June 20, 2003
- ☉ November 03-07 **5th International Symposium on Distributed Objects and Applications (DOA'2003)** Sicily, Italy. Topics include: Applications of distributed-object technology; Design patterns for object-based components and applications; Interoperability between object systems and complementary technology; Real-time solutions for distributed objects; Scalability for distributed objects and object middleware; Security for distributed-object systems; Software engineering for distributed object-based applications; Technologies for reliable and fault-tolerant distributed objects; etc. Deadline for submissions: June 15, 2003
- November 17-20 **14th IEEE International Symposium on Software Reliability Engineering (ISSRE'2003)** Denver, CO, USA. Deadline for submissions: May 1, 2003 (tutorials, industry abstracts, panels), August 1, 2003 (student papers, fast abstracts)
- November 19-21 **6th IFIP International Conference on Formal Methods for Open Object-based Distributed Systems (FMOODS'2003)** Paris, France. Deadline for submissions: June 2, 2003 (abstracts), June 9, 2003 (papers)
- November 19-21 **4th International IFIP Working Conference on Distributed Applications and Interoperable Systems (DAIS'2003)** Paris, France. In conjunction with FMOODS'2003. Deadline for submissions: June 2, 2003 (full papers), June 15, 2003 (work-in-progress papers)
- November 24-26 **2nd International Conference on Software for Embedded Systems (ICSTEST-E)** Bilbao, Spain. Topics include: Transportation and Safety-Critical Systems, Industry real experiences, Verification and Validation, Techniques for real time systems, Static and Dynamic analysis, etc. Deadline for submissions: April 15, 2003
- December 07-11 **2003 ACM SIGAda Annual International Conference (SIGAda'2003)** San Diego, California, USA. Sponsored by ACM SIGAda (ACM approval pending). In Cooperation With Ada-Europe and ACM SIGAPP, SIGCAS, SIGCSE, SIGPLAN, and SIGSOFT (Cooperation approvals pending). Topics include: Reliability needs and styles; Safety and high integrity issues; Improvements and additions to Ada to be included in Ada 200Y; Use of the Ada Distributed Systems Annex; Process and quality metrics; Testing and validation; Standards; Use of ASIS for new Ada tool development; Mixed-language development; Ada in XML environments; Ada in .Net environments; Quality Assurance; Ada & software engineering education; Commercial Ada applications: what Ada means to the bottom line; Use of SPARK and Ravenscar profile for high reliability software; Use of Real-Time CORBA; Real-time networking/quality of service guarantees; Fault tolerance and recovery; Distributed system load balancing; Static and dynamic code analysis; Performance analysis; Debugging complex systems; Integrating COTS software components; System Architecture & Design; Information Assurance in the age of terrorism. Deadline for submissions: June 8, 2003
- December 10 Birthday of Lady Ada Lovelace, born in 1815. Happy Programmers' Day!
- December 10-12 **European Software Process Improvement Conference (EuroSPI'2003)** Warwick, UK. Deadline for submissions: May 9, 2003
- December 10-12 **10th Asia-Pacific Software Engineering Conference (APSEC'2003)** Chiang Mai, Thailand. Topics include: Object-Oriented Technology and Patterns, Software Components, Programming Languages and Software Engineering, Reliability and Security, Formal Methods, Software

Development Environments, Distributed and Parallel Systems, etc. Deadline for submissions: June 16, 2003 (abstracts), June 30, 2003 (papers)

2004

- January 01/05-08 Software Technology Track of the 37th **Hawaii International Conference on System Sciences (HICSS-37)** Big Island of Hawaii, USA. Includes mini-tracks on: Distributed Object and Component-based Software Systems (Design Patterns for Distributed Systems, Middleware, Programming Languages and Environments for Distributed Object and Component Systems, ...); etc.
- January 03/29-April 02 **European Joint Conferences on Theory and Practice of Software (ETAPS'2003)** Barcelona, Spain. Event includes: conferences from 29 March to 2 April, 2004, affiliated workshops on 27-28 March and 3-4 April, 2004
- January April 05-07 4th **International Conference on Integrated Formal Methods 2004 (IFM'2004)** Canterbury, Kent, England. Deadline for submissions: September 15, 2003

8th International Conference on Reliable Software Technologies



Ada-Europe 2003

Toulouse, France, June 16-20, 2003

<http://www.ada-europe.org/conference2003.html>



In 2003, the 8th International Conference on Reliable Software Technologies will take place in Toulouse, France, from 16 – 20 June. The conference offers a technical program and exhibition, plus a series of tutorials and a workshop.

The conference provides an international forum for researchers, developers and users of reliable software technologies. Presentations and discussions cover applied and theoretical work currently carried to support the development and maintenance of software systems.

The program includes papers by the international community with refereed contributions from many countries. The proceedings of the conference will be published in the Lecture Notes in Computer Science (LNCS) Series by Springer. An exhibition concurrent to the conference offers an opportunity to explore the latest developments on the commercial marketplace. The tutorials on Monday and Friday offer an excellent opportunity to obtain in-depth knowledge in important technologies in the field.

With around 400 000 inhabitants, Toulouse is the fourth largest metropolitan area in France and the main city of the Midi-Pyrénées region, situated in the South-West of France.

Overview of the Week

	Morning	Late Morning	After Lunch	Afternoon
Monday June 16th Tutorials	The Personal Software Process (sm) for Ada Daniel Roy			
	Developing High Integrity Systems with GNAT/ORK Juan Antonio de la Puente & Juan Zamorano			
	Implementing Design Patterns in Ada 95 Matthew Heaney		Principles of Physical Software Design in Ada 95 Matthew Heaney	
Tuesday June 17th Sessions	An Invitation to Ada 2005 Invited talk P. Leroy	Ravenscar	Static Analysis	Ada-Europe General Assembly
		Language Issues	Vendor presentations	
Wednesday June 18th Sessions	Aspect-Oriented Programming beyond Hierarchical Modularity Invited talk M. Mezini	Distributed Information Systems	Software Components	Formal Specification
		Metrics	Vendor presentations	Vendor presentations
Thursday June 19th Sessions	Software Fault Tolerance Invited talk J. Kienzle	Real-Time kernel	Real-Time Systems	Closing Session & Awards
		Testing	Vendor presentations	
Friday June 20th Tutorials & Workshop	High Integrity Ravenscar using SPARK Peter Amey		Architecture Centric Development using Ada and the Avionics Architecture Description Language Bruce Lewis	
	A Semi Formal Approach to Software Systems Development William Bail		An Overview of Statistical-Based Testing William Bail	
	Workshop: Quality of Service (QoS) in Component-Based Software Engineering (CBSE)			



Technical Papers



Transactions and Groups as Generic Building blocks for Software Fault Tolerance

Marta Patiño-Martínez, Ricardo Jimenez-Peris & Alexander Romanovsky

Adding Temporal Annotations and Associated Verification to Ravenscar Profile

Alan Burns & Tse-Min Lin, UK

High Integrity Ravenscar

Peter Amey & Brian Dobbing, UK

A Graphical Environment for GLADE

Ernestina Martel-Jordán, Francisco Guerra-Santana, Javier Miranda-González, & Luis Hernández-Acosta, Spain

Experiences on Developing and Using a Tool Support for Formal Specification

Tommi Mikkonen, Finland

Booch's Ada vs. Liskov's Java: Two Approaches to Teaching Software Design

Ehud Lamm, Israel

A Case Study in Performance Evaluation of Real-Time Teleoperation Software Architecture using UML-MAST

Francisco Ortiz, Bárbara Álvarez, Juan A. Pastor & Pedro Sánchez, Spain

A Behavioural Notion of Subtyping for Object-Oriented Programming in SPARK95

Tse-Min Lin & John A. McDermid, UK

Running Ada on Real-Time Linux

Miguel Masmano, Jorge Real, Ismael Ripoll & Alfons Crespo, Spain

A Test Environment for High Integrity Software Development

Alejandro Alonso, Juan Antonio de la Puente & Juan Zamorano, Spain

The Standard UML-Ada Profile

Francis Thom & Emma Lines, UK

Busy Wait Analysis

Johann Blieberger, Bernd Burgstaller & Bernhard Scholz, Austria

Getting System Metrics Using POSIX Tracing Services

Agustín Espinosa Minguet, Vicente Lorente Garcés, Ana García-Fornes & Alfons Crespo i Lorente, Spain

Quasar : A New Tool for Concurrent Ada Programs Analysis

S. Evangelista, C. Kaiser, J.F. Pradat-Peyre & P.

Rousseau, France

Testing Safety Critical Ada Code Using non Real-Time Testint

Y.V. Jeppu, K. Karunakar & P.S. Subramanyam, India

Evidential Volume Approach for Certification

Silke Kuball & Gordon

Hughes, UK

A Quality Model for the Ada Standard Container Library

Xavier Franch & Jordi Marco, Spain

Some Architectural Features of Ada Systems Affecting Defects

William Evanco & June Verner, USA

Normalized Restricted Random Testing

Kwok Ping Chan, Tsong Yueh Chen & Dave Towey,

Australia - Hong-Kong

The Use of Ada, GNAT.Spitbol and XML in the Sol-Eu-Net Project

Mário Amado Alves, Alípio Jorge & Matthew Heaney, Portugal - USA

A Comparison of the Asynchronous

Transfer of Control Features in Ada and the Real-Time Specification for Java

Benjamin M. Brosgol & A.J. Wellings, UK

Towards Static Verification of Real-Time Performance: The Case of the GOCE

Platform Application Software

Niklas Holsti & Thomas Långbacka, UK

A Round-Robin Scheduling Policy for Ada

A. Burns, M. Gonzalez Harbour & A. Wellings, Spain -

UK

A Proposal to Integrate the POSIX

Execution-Time Clocks into Ada95

J. Miranda & M. González Harbour, Spain

A Survey of Physical Unit Handling

Techniques in Ada

C. Grein, Germany

Exposing Memory Corruptions and

Plumbing Leaks: Advanced Mechanisms in Ada

Emmanuel Briot, Franco Gasperoni, Robert Dewar,

Philippe Waroquiers & Dirk Craeynest, Belgium - France - USA

HRT-UML: Taking HRT-HOOD onto UML

Silvia Mazzini, Massimo D'Alessandro, Marco Di Natale,

Andrea Domenici, Giuseppe Lipari & Tullio Vardanega, Italy

Charles: A Data Structure Library for Ada 95

Matthew Heaney, USA

Eliminating Redundant Range Checks in

GNAT Using Symbolic Evaluation

Bernd Burgstaller, Austria

The above information is preliminary – please refer to the conference web site for the latest details.

<http://www.ada-europe.org/conference2003.html>

	<p style="text-align: center;">12th International Real-Time Ada Workshop IRTAW-12 15-19 September 2003 Pousada Monte de Sta. Luzia, Viana do Castelo Portugal</p>
---	---

Last Announcement before Close of Submission Period

Over the last 15 years, the series of the **International Workshop on Real-Time Ada Issues** have provided focus for identifying issues with Ada 83 and 95; for proposing solutions to those problems; and for evaluating proposed changes to the language standard.

Since the standardization of Ada 95, the **IRTAW** series have assisted in the review of the real-time related chapters of the *Guide for the Use of the Ada Programming Language for High Integrity Systems* (ISO/IEC TR 15942:2000) and have developed and promoted the Ravenscar Tasking Profile. With the advent of Java and the development of Real Time Specification for Java, the Workshop has begun to consider the integration of embedded Ada and Java systems, and their interoperability.

In keeping with this tradition, the goals of **IRTAW-12** will be to:

- Examine the shape and the status of the language amendments proposed or promoted by previous **IRTAW** editions, with respect to the ongoing Ada language revision process, as well as the demand for further enhancements
- Consider the lessons learned from industrial experience with using Ada 95 in general and the Ravenscar Profile in particular in actual real-time projects
- Examine and develop paradigms for using Ada 95 for real-time distributed systems, including issues of robustness as well as of hard, flexible and application-defined scheduling
- Consider the impact of the OOP paradigm on multi-threaded, possibly distributed, real-time systems
- Review the status and contents of the *Guide for the Use of the Ada Ravenscar Profile in High Integrity Systems* (to become an ISO/IEC TR) and examine the issues of certifying software systems developed using Ada 95
- Examine the status of and the progress with the Real-Time Specification for Java and consider user experience with the reference implementation and with issues of interoperability with Ada in embedded real-time systems.

Participation at **IRTAW-12** is by invitation following the submission of a position paper addressing one or more of the above topics. Position papers should not exceed six pages. All accepted papers will appear, in their final form, in the Workshop Proceedings, which will be published as a special issue of Ada Letters (ACM Press).

Submit position papers, in PDF, to the Program Chair by e-mail at: tullio.vardanega@math.unipd.it

Program Committee:

Ben Brosgol, Alan Burns, Brian Dobbing, Michael Gonzalez Harbour, Mike Kamrad, Stephen Michell, Luis Miguel Pinho (Local Chair), Juan Antonio de la Puente, George Romanski, Joyce Tokar, Tullio Vardanega (Program Chair), Andy Wellings.

Important Dates

Receipt of Position Paper: 1 June 2003

Final Copy of Paper: 1 November 2003

Notification of Acceptance: 30 June 2003

Workshop Date: 15-19 September 2003



<http://www.hurray.isep.ipp.pt/irtaw2003>



A Guided Tour to the GNAT Run-Time

Javier Miranda

*Applied Microelectronics Research Institute. University of Las Palmas de Gran Canaria, Canary Islands, Spain.
email: jmiranda@iuma.ulpgc.es <http://www.iuma.ulpgc.es/users/jmiranda>*

Abstract

This paper presents a free book which describes the GNAT implementation of the Ada Run-Time. In order to facilitate the use of the book for research and teaching, the html version of the book has hyperlinks to the GNAT sources and the Annotated Ada Reference Manual. This allows the reader to verify the contents of the book and a fast access to analyse the details of the Ada Run-Time not described in the book. Additionally, most of the papers referenced to in the bibliography section are available by means of hyperlinks.

Keywords: *Ada, Run-Time, GNARL, GNAT, GPL*

1 Introduction

It is well known that free software needs free manuals: reference and user manuals [1]. This allows the general community to “use” the technology. However, free software has a tremendous potential for research and teaching: the sources permit us to “understand” the technology. Nowadays, due to the lack of free books which describe the behaviour of free technology, most researchers and educators interested in it must repeat the same unpleasant task: to read the sources to try to understand the architecture and details of the free software. Obviously this is time consuming and error prone. This is also the case of the free GNU Ada Compiler: GNAT.

GNAT is currently a mature technology used for many industrial projects, research, and education. However, the lack of free books which describe real implementation hinders the modification of the sources to many researchers.¹ The main benefits of a free book on GNAT are:

- **The book can be used and enhanced by many researchers.** If the book is free it can be used as a shared document which provides an easy way to learn about the Run-Time (thus reducing the time required to experiment with Ada), but the book can also be upgraded by any researcher of the Ada community.

¹ Until now a researcher who needed to modify the behaviour of the GNAT run-time had only two main sources: the well documented sources (written in Ada), and some out of date published papers and documents.

- **The book facilitates the use of Ada to teach students about compilers and real-time systems.** The sources of the GNAT run-time are very well commented and written in Ada, thus providing a high level of abstraction which is quite useful for teaching.

The main goals of this project are:

- **To provide documentation on the GNAT Run-Time.** The contents of the book is my knowledge on the GNAT run-time.
- **To keep the book fully integrated with the GNAT sources.** Although the book is distributed in several formats for printing, the goal is to write a digital book with many links to the GNAT sources which facilitate the understanding and verification of the main concepts of the Ada run-time.
- **To keep the document free.** The book is distributed under the GNU Free Documentation License [2]. This allows members of the Ada community to enhance the quality of the book for research and teaching.

This paper is structured as follows: Section 3 presents the technical contents of the book. Section 4 deals with the digital version of the book. Section 4 gives the URL to access the book and describes the available formats. Section 5 provides the background of this work. Section 6 presents the main lines of continuation of this work. The paper concludes with a short biography of the author and some references.

3 Technical Contents

In order to facilitate the reading of the book for teaching, each chapter is structured in two parts. The first part has a brief summary of the Ada concepts described in the chapter (I used the excellent book *Concurrency in Ada*[3] as a reference to write these summaries). The second part of the chapter describes GNAT implementation of the concepts presented in the first part. Currently the book has the following chapters:

1. **The GNAT project.** Brief introduction to the GNAT project, and the overall architecture of the compiler and run-time.
2. **Task types and objects.** This chapter describes the main aspects of the life-cycle of Ada tasks: task creation, task activation, task termination, and task identification.

3. **The Rendezvous.** This chapter deals with the handling of the entry call parameters, the rendezvous queues and the basic rendezvous modes (simple, conditional and selective).
4. **Protected Objects.** The Ada95 eggshell model for protected objects and its implementation is presented here. Several alternative implementations are also described and discussed.
5. **Time and clocks.** This chapter covers the Ada timed sentences: timed entry call and timed selective accept.
6. **Interrupts.** The Ada model of interrupts and its implementation is presented here.
7. **Exceptions.** Data types used to identify the exceptions, and hash table used by GNAT to handle the exceptions are presented here.
8. **Abort and Asynchronous Transfer of Control.** Ada tasks abortion and the implementation of the Ada95 asynchronous transfer of control are discussed in this chapter.
9. **Bibliography.** This chapter provides the bibliography used to write the book.

4 The digital book

This section provides a brief guided tour of the digital version of the book. This allows the reader to see the proposed use of the book for teaching and research.

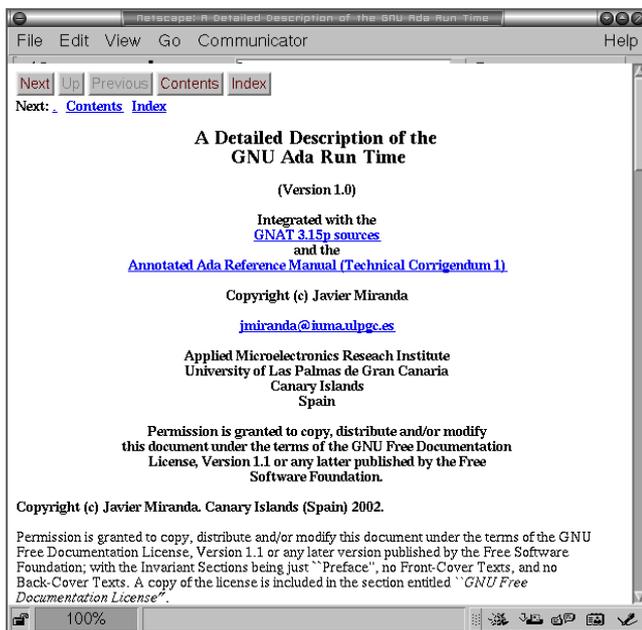


Figure 1: The main page of the book

We will start with the main page (c.f. Figure 1). In addition to the book title and copyright, this page has two hyperlinks: access to the GNAT 3.14p sources main index and access to the Annotated Ada Reference Manual (Technical Corrigendum 1) [6].

The index of the GNAT 3.14p sources was automatically generated with the `gnat2html.pl` Perl program which comes with the GNAT 3.14p binary distribution (c.f. Figure 2). This index gives us the ordered list of source files and

subprograms, fast access to them, and a coloured presentation of the GNAT sources.

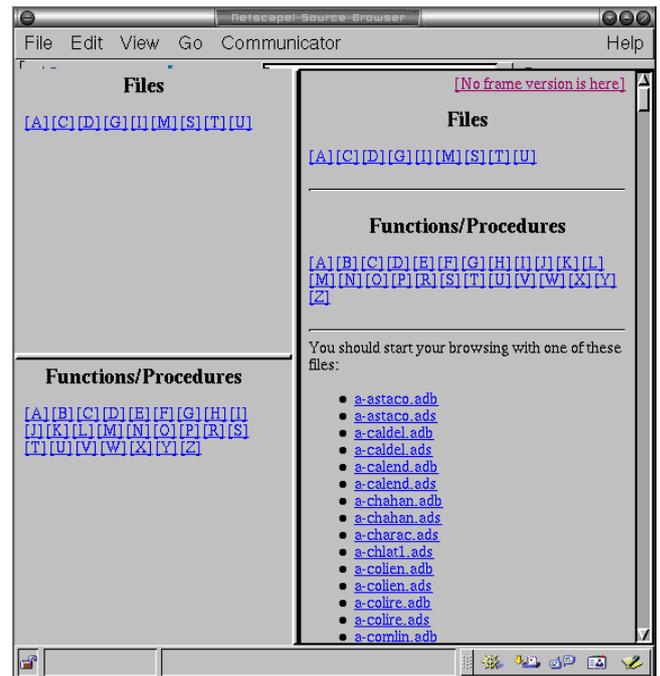


Figure 2: Index to the GNAT sources

From the table of contents of the book (c.f. Figure 3) we have a direct access to any section of the book.

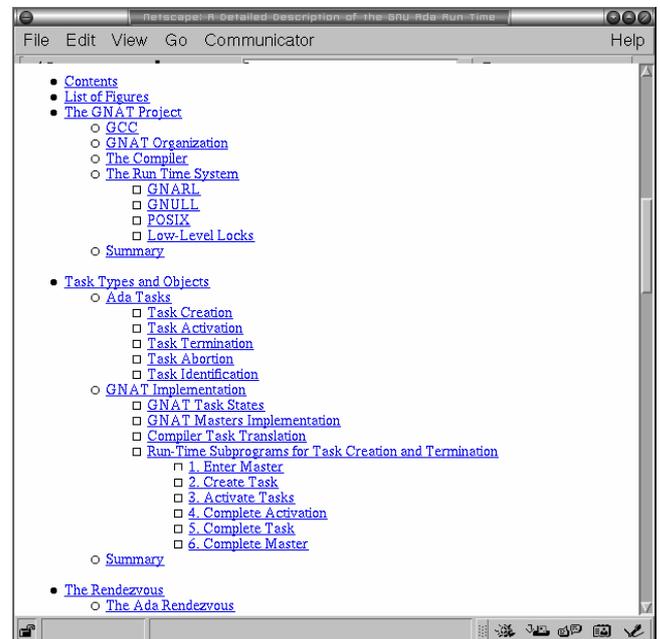
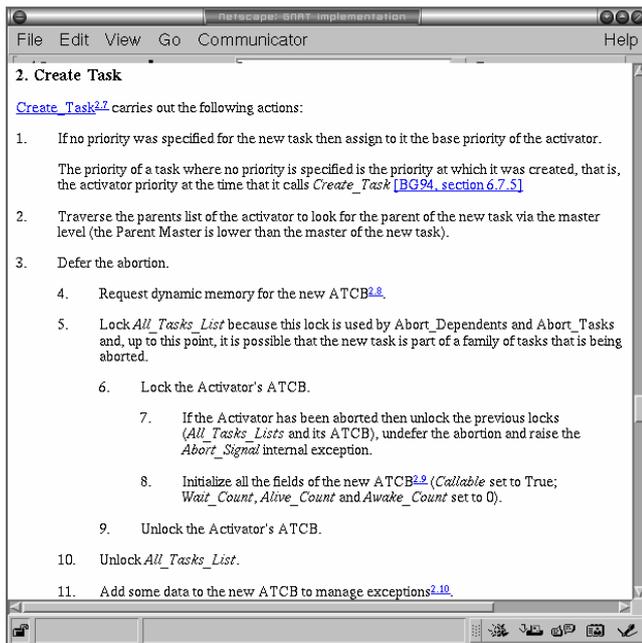


Figure 3: Table of contents

As an example, let's choose the reading of the implementation of tasks creation (c.f. Figure 4).

Figure 4: Create task description



This page describes the main actions performed by the GNAT run-time for tasks creation. We can verify these actions by clicking on the hyperlink (Create_Task) to jump to the GNAT source file where this subprogram is implemented (c.f. Figure 5).

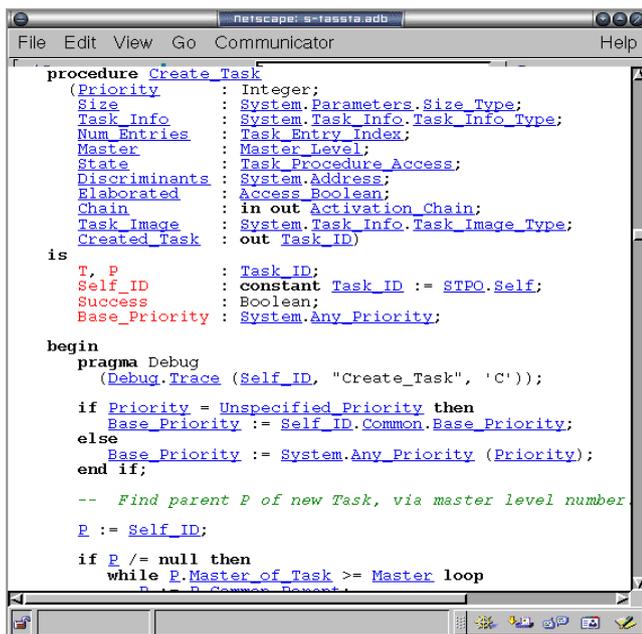


Figure 5: Create task implementation

In addition to the generation of the source index, the tool `gnathtml.pl` also links all the source lines sources with their corresponding object and type declarations. This allows the reader to traverse the sources with a preferred web navigator, and facilitates the understanding of GNAT Run-Time.

Finally, in the bibliography section, the book has hyperlinks to the NEC Research Index (c.f. Figure 6) which

allows the reader to obtain a copy of most of the referenced documents.

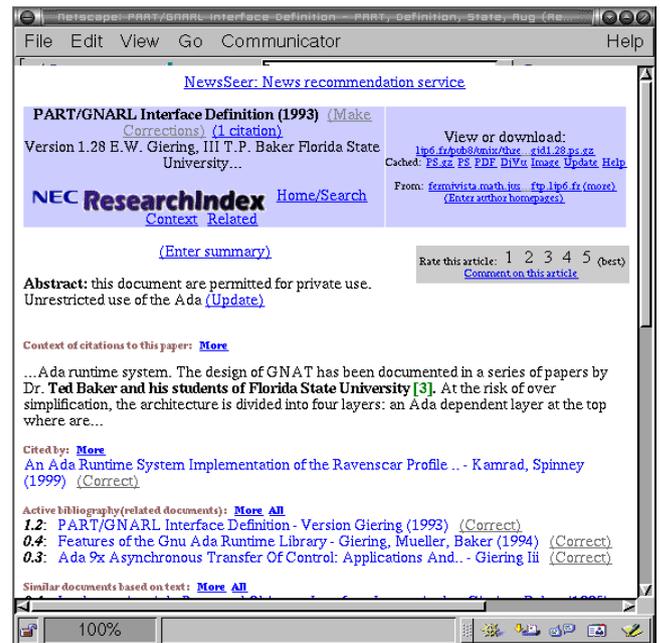


Figure 6: NEC Research Index

4 Distribution

The book is available at:

<http://gnat.webhop.info>

<http://www.iuma.ulpgc.es/users/jmiranda/>

The first address provides an “easy to remember” access to the second address, which is the real address of my web page. Nowadays the distribution of the book includes:

- The **html** version of the book. The book is always available by internet as an on-line book (accessible to your preferred web navigator).
- A compressed **html.tgz** file with the html sources. This distribution includes one script which patches the html files and installs them locally (in any local directory) or in your personal web page.
- The **PostScript**, **PDF**, and **DVI** files of the whole book. This distribution facilitates the printing of the book. In order to not lose information, most of the html distribution hyperlinks are included in this distribution by means of footnotes.
- The **LaTeX sources** of the book. This distribution allows the members of the Ada community to cooperate to improve the quality of the document.

Although it is well known that the GNU format for free books is Texinfo, the book is written in LaTeX. The main reasons are:

- I do not know Texinfo and I feel quite familiar with LaTeX.

- LaTeX is well known to the scientific community. This facilitates any member of the Ada community to modify the sources of the book to improve its contents.
- LaTeX is also free software. It is obvious that a free-book must be written with free software to avoid any future problem with the distribution of the sources (due to the copyright of the file formats).
- LaTeX2html is a good tool to make the automatic translation of the book to HTML. Therefore it is not necessary to maintain two versions of the book (one for the printed version, and another for the HTML files).

5 Background

A previous version of this book was written in 1999 under the title “*How to modify the GNAT Run-Time to experiment with Ada extension*” [4]. That book was the result of a project to integrate Drago[5] into the GNAT sources. (Drago was the result of my PhD research. It is an Ada extension which facilitates the programming of fault-tolerant and cooperative distributed applications by means of the addition of the groups paradigm into the Ada language).

Although the book was available on the web, no publicity was sent out. Ingenuously we thought that the internet search engines would provide the book to any interested people. Obviously very few people found it, but we thought that the book was not good enough and we stopped our efforts.

During the AdaEurope’2001 conference (Belgium) I presented the book to some colleagues and felt that the book could still be of interest for the Ada community. Therefore, in July 2001 I decide to personally restart the project with the following goals:

- **Write the manuscript in English.** The manuscript of the previous version was written in Spanish and translated to English. Because I was now alone to do the whole work, I decided to concentrate any effort in the most widely distributable version of the book. Once completed many people can help to translate it to other languages.
- **Recover the (still) useful documentation of the previous book.** Although most of the chapters of the previous book were now obsolete, some parts were still reusable (the previous book described the GNAT-3.10p sources).
- **Structure each chapter of the book in two parts.** The first part summarizes the Ada concepts whose implementation is described in the second part. This will facilitate the use of the book for teaching.
- **Reduce the maintenance cost as much as possible.** A single document should be used for all the distributions of the document. I decided to write the sources of the book with LaTeX because it is free and has many free

tools to translate the document to HTML, PostScript, PDF, and DVI files.

- **Keep the book integrated with the GNAT sources.** The HTML version of the book would be the “star” of the project with hyperlinks which facilitate to the reader the direct access to the Annotated Ada Reference Manual [6] and GNAT sources.

I have been working on the book for one year. In July 2002 the first Beta version of the book was available in the WEB. In September 2002 I added the GNU Free Documentation License to the book, and I put the full sources on my web site.

6 Future work

There are two main lines of continuation of this work:

1. **To upgrade the current chapters.** Although I propose to keep the document upgraded, many experts in this field could help me to enhance and complete the book with many additional details². From here, if you are this person, I welcome your help!
2. **Add new chapters.** Although the book currently covers most of the chapters required for teaching the Ada runtime, many additional aspects should be described (i.e. controlled types, Ada annexes, etc).

This will allow us to achieve the high quality desired for this kind of document.

7 Acknowledgements

I would like to thank professor Edmond Schonberg for replying to me many messages with questions on the GNAT compiler. I also thank professor Ted Baker for offering his help to enhance future versions of this book.

I acknowledge Alexis González and José Jerónimo Martín for the tremendous effort done during the previous project (which was the base of this book). I also thank my colleagues of the Distributed Systems Research Group (University of Las Palmas de Gran Canaria) Francisco Guerra, Ernestina Martel, José Miguel Santos, and Luis Hernández for the discussions which have helped me to understand many details of the GNAT sources. Finally I also thank David Shea (Faculty of Translation and Interpreting, University of Las Palmas de Gran Canaria) for all the corrections done to improve the English version of this paper.

² It is important to avoid the description of GNAT aspects which are still under development and change very fast. They are found by comparison of sources of successive GNAT distributions.

Short Biography

Javier Miranda was born in 1965 in Canary Islands (Spain). He studied Computer Science Engineering at the University of Las Palmas de Gran Canaria. He finished his studies by implementing a Modula-2 compiler under the direction of José Fortes Gálvez. In 1990 he went to the Polytechnic University of Madrid (Telematics Engineering Department ---DIT) to do his PhD research under the direction of Angel Álvarez and Sergio Arévalo (in the Distributed Systems Research Group). In 1994 he finished his PhD thesis entitled "*Drago: A Language for Programming Fault-Tolerant and Cooperative Distributed Applications*". In 1997, he started a project to integrate Drago into the GNAT sources. This project was done in collaboration with his colleague Francisco Guerra (who implemented the required protocols in the Group_IO Ada Library), and the students Alexis Rodríguez and José Jerónimo Martín. After the project he personally continued upgrading Drago to the next GNAT distributions. This book summarizes the experience achieved during this project.

References

- [1] [R. Stallman. Free Software and Free Manuals. Essay. <http://www.gnu.org/philosophy/free-doc.html>
- [2] [2] GNU. GNU Free Documentation License. <http://www.gnu.org/copyleft/fdl.html>. March, 2000.
- [3] A. Burns, A. Wellings. Concurrency in Ada (2nd edition). Cambridge University Press, 1998.
- [4] J. Miranda, F. Guerra, A. Gonzalez, J. Martín. How to modify the GNAT Run Time to experiment with Ada extensions. University of Las Palmas de Gran Canaria, Canary Islands, Spain. ISBN: 84-87526-68-3. Available at: <http://www.iuma.ulpgc.es/users/gsd/Drago>
- [5] J. Miranda. Drago: Reference Manual, 1994. Available at: <http://www.iuma.ulpgc.es/users/gsd>
- [6] Annotated Ada Reference Manual (revised edition that includes all of the corrections from Technical Corrigendum 1). Available at: <http://www.ada-auth.org/~acats/arm.html>

Increasing the Real-Time Expressive Power of Ada 95

Alan Burns

Department of Computer Science, University of York, York YO10 5DD, UK. Email: burns@cs.york.ac.uk

Abstract

As outlined in an article appeared in the previous issue of this journal, a window of opportunity has recently arisen for the Ada community to put forward language enhancements that strengthen the position of Ada in the market segments where it already leads and that make it more competitive in other attractive areas. The IRTAW (International Real-Time Ada Workshop) group, which represents a sizeable proportion of the leading real-time Ada community has seized this opportunity and started to submit proposals for specific language enhancements, for which both the user expectation and the underlying methodological aspects are mature. In this short paper we outline some of the proposed new real-time features of the language.

1 Introduction

Some of the most important changes made to Ada during the 9X process concerned the concurrent and real-time aspects of the language. The introduction of protected objects and asynchronous transfer of control significantly improved the expressive power of the tasking model. The introduction of a set of features in the Real-Time Annex similarly provided a sound basis for the support of real-time applications. Now we have the opportunity to make some minor modifications to the language definition, which enable us to again extend the real-time features of Ada. In an article in the last issue of the Ada User Journal [1] Pascal Leroy, the chair of the Ada Rapporteur Group (ARG), delegated authority for the language revision process, outlined a general overview of the modification process; in this paper we concentrate on the features being considered that support real-time programming.

We split this presentation into three parts.

- A discussion of those features already agreed – the Ravenscar Profile and a new Partition Elaboration Policy.
- A discussion of those features under detailed consideration – Execution Time Budgeting and Timing Events.
- A short outline of other features that await to be evaluated as part of the amendment process.

2 Ravenscar and partition elaboration policy

There is increasing recognition that the software components of critical real-time applications must be provably predictable. This is particularly so for a hard real-time system, in which the failure of a component of the system to meet its timing deadline can result in an unacceptable failure of the whole system. The choice of a suitable design and development method, in conjunction with supporting tools that enable the real-time performance of a system to be analysed and simulated, can lead to a high level of confidence that the final system meets its real-time constraints.

The Ravenscar Profile [2] is a subset of the tasking model, restricted to meet the real-time community requirements for determinism, schedulability analysis and memory-boundedness, as well as being suitable for mapping to a small and efficient run-time system that supports task synchronization and communication, and which could be certifiable to the highest integrity levels. The concurrency model promoted by the Ravenscar Profile is consistent with the use of tools that allow the static properties of programs to be verified. Potential verification techniques include *information flow analysis*, *schedulability analysis*, *execution-order analysis* and *model checking*. These techniques allow analysis of a system to be performed throughout its development life cycle, thus avoiding the common problem of finding only during system integration and testing that the design fails to meet its non-functional requirements.

It is important to note that the Ravenscar Profile is silent on the non-tasking (i.e. sequential) aspects of the language. For example it does not dictate how exceptions should, or should not, be used. For any particular application, it is likely that constraints on the sequential part of the language will be required. These may be due to other forms of *static analysis* to be applied to the code, or to enable *worst-case execution time* information to be derived for the sequential code. The Ravenscar Profile has been designed such that the restricted form of tasking that it defines can be used even for software that needs to be verified to the very highest integrity levels.

The 8th International Real-Time Ada Workshop (IRTAW) was held in April 1997 at the small Yorkshire village of Ravenscar. Two position papers [3][4] led to an extended discussion on tasking profiles. By the end of the workshop, the Ravenscar Profile had been defined [5] in a form that is almost identical to its current specification.

At the 9th IRTAW [6] (March 1999) the Ravenscar Profile was again discussed at length. The definition was reaffirmed and clarified. The most significant change was the incorporation of Suspension Objects. An Ada Letters paper [5] became the de facto defining statement of the Profile.

By the 10th IRTAW [7] (September 2000) many of the position papers were on aspects of the Profile and its use and implementation. No major changes were made although an attempt to standardize on the Restriction identifiers was undertaken. Time was spent on a non-preemptive version of the Profile. Following the 10th workshop the participants decided to forward the Ravenscar Profile to the ARG – the ISO body in charge of the maintenance of the Ada language – so that its definition could move from a de facto to a real standard.

At the 11th IRTAW [8] (April 2002) the formal definition of the profile as formulated by the ARG was agreed. It was also confirmed that the Profile requires task dispatching policy `FIFO_Within_Priorities` and locking policy `Ceiling_Locking`.

The HRG – the ISO body in charge of the high integrity aspects of the Ada language – was tasked with producing a Rationale for the Profile. In response to this action, a detailed guide on the Profile, its definition, justification and use, has recently been produced and is available online [2].

In outline, the definition uses the Restrictions pragma to restrict the tasking and real-time features that can be used when profile Ravenscar is defined for a partition. To do this some new Restriction identifiers have had to be introduced. – see [2] for the details.

Closely coupled to Ravenscar is a new pragma, `Partition_Elaboration_Policy(Sequential)`, which changes the language model that defines how tasks are activated in library packages. To satisfy the requirements of the safety critical and high integrity domains there is a need to postpone task activation and interrupt attachment until the Environment task has completed the elaboration of all library items and is ready to call the main subprogram. At this time all postponements are undertaken. This more deterministic model has a number of advantages, and for most (error free) programs has identical external behaviour to the normal model.

3 Features under detailed consideration

There are a number of language additions under detailed consideration by the ARG. Most important is a new library package for execution time budgeting. This allows a task's execution time to be monitored and budgets to be set on execution time. A protected type is defined in the package that will allow a task to block on an entry until such time as another designated task has used up its budget. This can be used to either catch an error (budget overrun) or as part of a real-time algorithm such as an imprecise algorithm, where an iterative process refines a result until a given budget is exhausted.

The package to support control over execution time may have a form similar to the following [9]:

```
with Ada.Task_Identification;
package Ada.Real_Time.Execution_Time is

  type CPU_Time is private;
  CPU_Time_First : constant CPU_Time;
  CPU_Time_Last : constant CPU_Time;
  CPU_Time_Unit : constant := implementation-
    defined-real-number;
  CPU_Tick : constant Time_Span;

  function Clock (T : Ada.Task_Identification.Task_ID
    := Ada.Task_Identification.Current_Task)
    return CPU_Time;

  function "+" (Left : CPU_Time; Right : Time_Span)
    return CPU_Time;
  function "+" (Left : Time_Span; Right : CPU_Time)
    return CPU_Time;
  function "." (Left : CPU_Time; Right : Time_Span)
    return CPU_Time;
  function "-" (Left : CPU_Time; Right : CPU_Time)
    return Time_Span;
  function "<" (Left, Right : CPU_Time)
    return Boolean;
  function "<=" (Left, Right : CPU_Time)
    return Boolean;
  function ">" (Left, Right : CPU_Time)
    return Boolean;
  function ">=" (Left, Right : CPU_Time)
    return Boolean;

  procedure Split (
    T : CPU_Time;
    SC : out Seconds_Count;
    TS : out Time_Span);

  function Time_Of (
    SC : Seconds_Count;
    TS : Time_Span)
    return CPU_Time;

  procedure type Timer (
    T : access Ada.Task_Identification.Task_ID) is

    procedure Arm (Interval : Time_Span);
    procedure Arm (Abs_Time : CPU_Time);
    procedure Disarm;

    entry Timer_Expired;

    function Timer_Has_Expired
      return Boolean;
    function Timer_Remaining
      return Time_Span;

  private
    ... -- not specified by the language
  end Timer;

  Timer_Error : exception;
  -- may be raised by Timer.Timer_Expired,
  -- Timer.Time_Has_Expired, and
  -- Timer.Time_Remaining
  Timer_Resource_Error : exception;
  -- may be raised on the declaration of a Timer or calls
  -- to either Arm
```

```
private
... -- not specified by the language
end Ada.Real_Time.Execution_Time;
```

Execution time or CPU time of a given task is defined as the time spent by the system executing that task, including the time spent executing run-time or system services on behalf of it. The mechanism used to measure execution time is implementation defined. It is left to the implementation to define to whom the execution time that is consumed by interrupt handlers and run-time services on behalf of the system will be charged.

Package `Execution_Time` contains a protected object type called `Timer`, which represents a software object capable of detecting execution time overruns. Each timer is attached to a specific execution time clock, which is the clock of the task specified by the access discriminant `T`. This protected object type has visible operations for the application tasks to arm or disarm a timer, and to determine whether a timer has expired or not (`Timer_Has_Expired`). In addition, `Timer` has an entry (`Timer_Expired`) that can be used by application tasks to block until an execution time overrun is detected, or as an event that triggers the abortion of the instructions of a select statement with an abortable part.

The reader is referred to §4.2 of [1] for an example of use of execution time clocks.

Another new feature that is being considered for the Real-Time Annex is the support of *timing events*.

An exploration of various flexible scheduling schemes, for example imprecise computation, has illustrated the need:

- to asynchronously change the priority of a task at a particular future time, and
- to allow tasks to come off the delay queue at a different priority to that in effect when the task was delayed.

This functionality can only be achieved by the use of a “minder” high-priority task that makes the necessary priority changes to its client. This is an inefficient and inelegant solution.

More generally, Ada provides only one mechanism for associating the execution of code with points in time. A lower level primitive would increase the applicability of the Ada language. A protected procedure can currently be associated with interrupt events; the proposal allows similar functionality for timing events. A possible form for the proposed facility is as follows:

```
package Ada.Real_Time.Timing_Events is

type Timing_Event is limited private;
type Timing_Event_Handler is access
protected procedure(Event : in out Timing_Event);

procedure Set_Handler(
Event : in out Timing_Event;
At_Time : in Time;
Handler: in Timing_Event_Handler);
```

```
procedure Set_Handler(
Event : in out Timing_Event;
In_Time: in Time_Span;
Handler: in Timing_Event_Handler);

function Is_Handler_Set(Event : Timing_Event)
return Boolean;
function Current_Handler(Event : Timing_Event)
return Timing_Event_Handler;

procedure Cancel_Handler(
Event : in out Timing_Event;
Cancelled : out Boolean);

function Time_Of_Event(Event : Timing_Event)
return Time;
```

```
private
... -- not specified by the language
end Ada.Real_Time.Timing_Events;
```

At a time no earlier than that implied by the time parameter, the `Handler` procedure is executed. The detailed rules governing timing accuracy are the same as those specified for the accuracy of the `Real_Time.Clock` required at clause D.9 of the Ada Language Reference Manual.

A call of either `Set_Handler` procedure is not a potentially suspending operation and hence can be called from within a `Timing_Event_Handler`.

A call to a `Set_Handler` procedure for a `Timing_Event` that is already set will initially cancel the earlier registration, while the latest one remains set. A number of timing events registered for the same time will execute in FIFO order. All will be executed before any other application code.

A typical example of usage is a watchdog timer. Here a condition is tested every 50 milliseconds. If the condition has not been called during this time an alarm handling task is released.

```
with Ada.Real_Time;
use Ada.Real_Time;
with Ada.Real_Time.Timing_Events;
use Ada.Real_Time.Timing_Events;
with System;
use System;
...
protected Watchdog is -- library level declaration
pragma Interrupt_Priority(Interrupt_Priority'Last);
entry Alarm_Control; -- called by alarm handling task
procedure Timer(TE : in out Timing_Event);
procedure Call_in; -- called by application code
-- every 50ms if alive

private
Alarm : Boolean := False;
end Watchdog;

Fifty_Mil_Event : Timing_Event;
TS : Time_Span := Milliseconds(50);

protected body Watchdog is

-- this entry is called by the alarm handling task
entry Alarm_Control when Alarm is
begin
```

```

Alarm := False;
end Alarm_Control;

procedure Timer(TE : in out Timing_Event) is
begin
  Alarm := True;
end Timer;

-- this procedure is called by application code
procedure Call_in is
begin
  Set_Handler(
    Fifty_Mil_Event,
    TS,
    Watchdog.Timer'access);
  -- this call to Set_Handler cancels the previous call
end Call_in;

end Watchdog;
...

```

4 Other features under consideration

The Real-Time Systems Annex in Ada95 defines a number of mechanisms for specifying scheduling policies. It also provides a complete definition of one such policy: `FIFO_Within_Priorities`. This policy, which requires preemptive priority based dispatching, is a natural choice for real-time applications. It can be implemented efficiently and leads to the development of applications that are amenable to effective analysis -- especially when combined with the immediate priority ceiling protocol (ceiling locking) on protected objects.

There are, however, application requirements that cannot be fully accomplished with this policy alone. An additional policy, `Non_Preemptive_FIFO_Within_Priority`, has already been discussed and agreed by the ARG. Round Robin dispatching is also being considered, and at the forthcoming 12th IRTAW a number of other schemes will be considered, including EDF scheduling. Another important aspect of scheduling is that of hierarchies of schedules; for example a priority based scheme that has Round Robin at some priority levels, EDF at others and FIFO in the rest. An increasing number of applications require this, and other forms of, partitioning of schedules to reflect partitioned subsystems within the overall application.

Another feature that is being considered is dynamic ceiling priorities for protected objects. Although dynamic priorities for tasks enable most dynamic scheduling algorithms to be programmed, there are some schemes (an example being mode changes) that require, for maximum efficiency, ceilings to be changed. This feature would allow that to happen.

5 Conclusion

The amendment to the language's definition is an opportunity to consolidate Ada's position as the most expressive language for programming real-time systems. An already extensive set of features can be extended with other well understood mechanisms to provide a modern and effective language that will support a wider range of applications, and challenge the research community to continue to put Ada at the heart of future developments.

References

- [1] P. Leroy, An Invitation to Ada 2005, *Ada User Journal* 23(4): 230-237.
- [2] A. Burns, B. Dobbing and T. Vardanega, Guide for the use of the Ada Ravenscar Profile in High Integrity Systems, University of York Technical Report, YCS 348, 2003. <ftp://ftp.cs.york.ac.uk/reports/YCS-2003-348.pdf>
- [3] A. Burns and A.J. Wellings, Restricted Tasking Models, *Ada Letters*, XVII (5), 27 - 32, 1997.
- [4] B. Dobbing. and M. Richard-Foy, T-SMART - Task Safe, Minimal Ada Realtime Toolset, *Ada Letters*, XVII (5), 45 - 50, 1997.
- [5] A. Burns, The Ravenscar Profile, *Ada letters*, XIX (4), 49 - 52, 1999.
- [6] Session Summary, The Ravenscar Profile and Implementation Issues, *Ada Letters*, XIX (2), 12 - 14, 1999.
- [7] Session Summary, Status and Future of the Ravenscar Profile, *Ada Letters*, XXI (1), 5 - 8, 2001.
- [8] Session Summary, Ravenscar Profile, Proceedings of the 11th International Real-Time Ada Workshop, *Ada Letters* (to appear in 2003).
- [9] J. Miranda Gonzalez, and M. Gonzalez Harbour, A Proposal to Integrate the POSIX Execution-Time Clocks into Ada 95. Technical Report, June, 2002.

Ada-Europe 2002 Sponsors

ACT Europe

Contact: *Zépur Blot*

8 Rue de Milan, F-75009 Paris, France

Tel: +33-1-49-70-67-16

Email: sales@act-europe.fr

Fax: +33-1-49-70-05-52

URL: www.act-europe.fr

Aonix GmbH

Contact: *Peter Dencker*

Durlacher Allee 95, D-76137 Karlsruhe, Germany

Tel: +49-721-98653-22

Email: dencker@aonix.de

Fax: +49-721-98653-98

URL: www.aonix.com

DDC International A/S (DDC-I A/S)

Contact: *Erik Jensen*

Gammel Lundtoftevej 1B, DK-2800 Lyngby, Denmark

Tel: +45-45-87-11-44

Email: eje@ddci.dk

Fax: +45-45-87-22-17

URL: www.ddci.com

Green Hills Software GmbH

Contact: *Hans-Joachim Fischer*

Unterreit 6, D-76135 Karlsruhe, Germany

Tel: +49-721-9862580

Email: sales-ger@ghs.com

Fax: +49-721-9862581

URL: www.ghs.com

RainCode Corp

Contact: *Deborah Torrekens*

1 Rue de l'Autonomie, B-1070 Bruxelles, Belgique

Tel: +32-2-522-06-63

Email: Deborah@raincode.com

Fax: +32-2-522-09-30

URL: www.raincode.com

Rational Software Austria GmbH

Contact: *Martina Rumpeltes*

Lassallestraße 7b, A-1020 Wien, Austria

Tel: +43-1-245-31-160

Email: info@rational.com

Fax: +43-1-245-31-200

URL: www.rational.com

TNI Europe Limited

Contact: *Pam Flood*

58A Mill Street, Congleton, Cheshire CW12 1AG, UK

Tel: +44-1260-29-14-49

Email: info@tni-europe.com

Fax: +44-1260-29-14-49

URL: www.tni-europe.com